API SPECIFICATIONS
FOR

HOSTED PAYMENT PAGE
+
3 ENDPOINTS
+
GENERAL OPERATIONAL FUNCTIONS

INTEGRATION
(Version 13.26)

Last updated on 27th Aug 2021

	RAZER MERCHANT SERVICES
	J-39-1, Block J, Persiaran Multimedia, i-City, 40000 Shah Alam, Selangor, Malaysia.
	+(603) - 5521 8438
	support-sa@razer.com
	merchant.razer.com
	Social Networks https://twitter.com/Razer_MS https://facebook.com/RazerMerchantServices https://youtube.com/c/RazerMerchantServices https://instagram.com/RazerMerchantServices https://linkedin.com/company/RazerMerchantServices
	Developer Platforms https://github.com/RazerMS Mobile XDK, seamless and inpage checkout, and many shopping carts payment plugin/module/addon/extension are available

ChangeLogs

2014/05/16	v11.0	Formalize the document
2014/06/14	v12.0	Simplified and consolidating API documents
2016/10/24	v13.0	Declare EOL of inactive channels
2016/12/03	v13.3	Introduce secret key and changed of some critical APIs hashing based on secret key
2017/09/08	v13.6	Removal of obsoleted channels
2017/10/10	v13.7	Splitting payment and non-payment request endpoint FQDN, adding sandbox URL
2018/05/31	v13.8	Adding Pick-n-Pay APIs
2018/08/20	v13.9	Adding card BIN info API
2019/01/04	v13.9	Update value of channel in return URL, withdraw Pay-n-Pick from this document
2019/04/30	v13.9	Rebranding to Razer Merchant Services
2019/05/07	v13.10	Withdrawal of escrow and mass payout APIs
2019/06/17	v13.11	Adding currency & error code/desc to requery APIs response
2019/09/20	v13.12	Transfering predefined bank and FTT list to Razer Escrow & Mass Payout API
2019/11/29	v13.13	Adding Taiwan payment channels, removing EOL channels table
2020/01/09	v13.14	Adding eWallet channel (GrabPay, Maybank QRPay), 51 currencies supported for multi-currency card acceptance
2020/03/24	v13.14	Update Settlement Report API version 3.0 to 4.0 that includes refund and chargeback records
2020/08/27	v13.15	Adding Thailand credit card channel.
2020/09/18	v13.16	Changing all API FQDN to Razer domain; Enhanced settlement report API to support duration in seconds from a specific time
2020/10/02	v13.16	Minor enhancement on the daily transaction report API to include all optional fields
2020/11/06	v13.17	Adding Cash-TH channel
2020/11/24	v13.18	Adding Cash-KKMart channel
2021/01/07	v13.19	Adding ShopeePay channel
2021/01/14	V13.20	Adding new FPX B2C & B2B channels (Agrobank, Citibank, UOB, Bank Rakyat, Affin, and etc.)
2021/01/29	V13.21	Minor enhancement on daily transaction report and Capture Request API
2021/03/10	v13.22	Adding/removing of payment channel's error code table
2021/06/09	v13.23	Adding DuitNow group channel, additional optional billing info, and update Thailand channel
2021/07/01	v13.23	Adding AppDeeplink parameter to payment request
2021/07/13	v13.24	Adding Singapore PayNow

2021/07/28	v13.24	Adding eNets Debit error code & removing Razer Pay
2021/08/24	v13.25	Changing all www.onlinepayment.com.my to pay.merchant.razer.com
2021/08/27	v13.26	Adding installment channels

Table of Contents

INTRODUCTION

Dealing with banks and multiple payment channels are time-consuming and it might cost you a lot of money. We also understand that it's a hassle for those non-technical people to integrate even one kind of online payment method into their existing website. Thus we consolidate all payment channels into one integration to reduce merchant's costs and effort, and to shorten your time-to-market.

Razer Merchant Services (RMS) is to help merchants, who sell online and offline, to reach out to the regional buyers, collecting payment and distributing funds, expanding rapidly to the Southeast Asia market. The service includes:

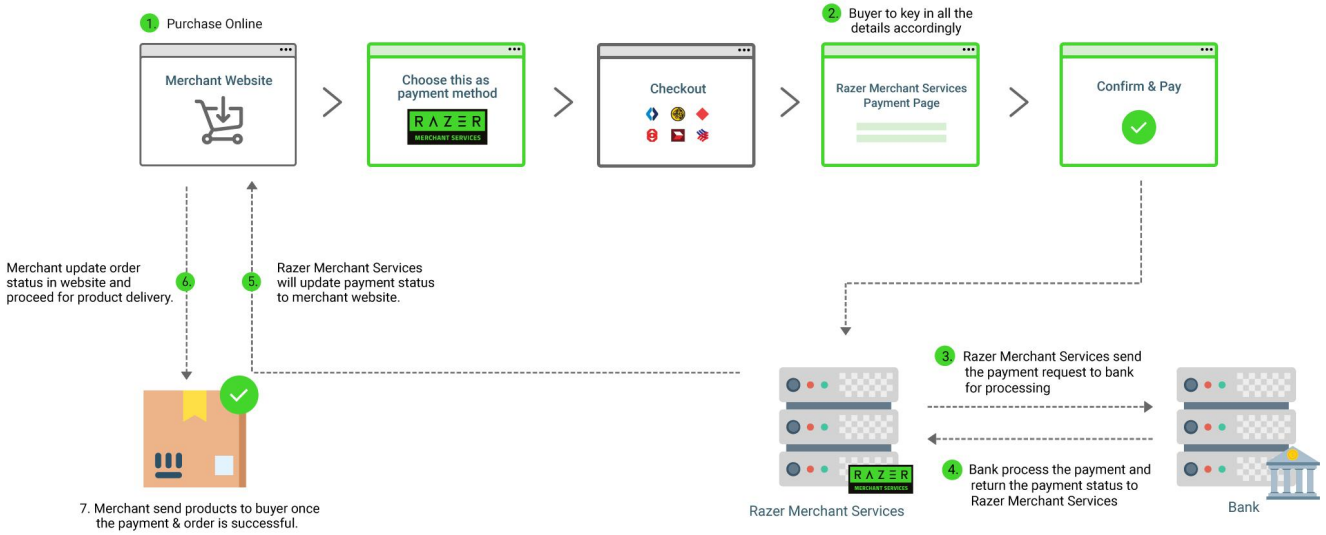
- Front-end
    - RWD or responsive web design payment page for online buyer to checkout
    - Secure 1-click payment using credit card tokenization technology, to ease the payment flow for returning purchase
    - Bcard redemption and rewarding loyalty program for buyers
    - Seamless integration to popup bank login page immediately
    - In-page checkout similar to Stripe and airbnb checkout flow that can seamlessly integrated with merchant system
    - Mobile XDK allows app developers to implement in-app payments
    - Channel switching is available for same currency channels
    - Razer Cash which allows over-the-counter (OTC) payment acceptance
    - Common shopping carts payment module, plugin, addon, or extension supported

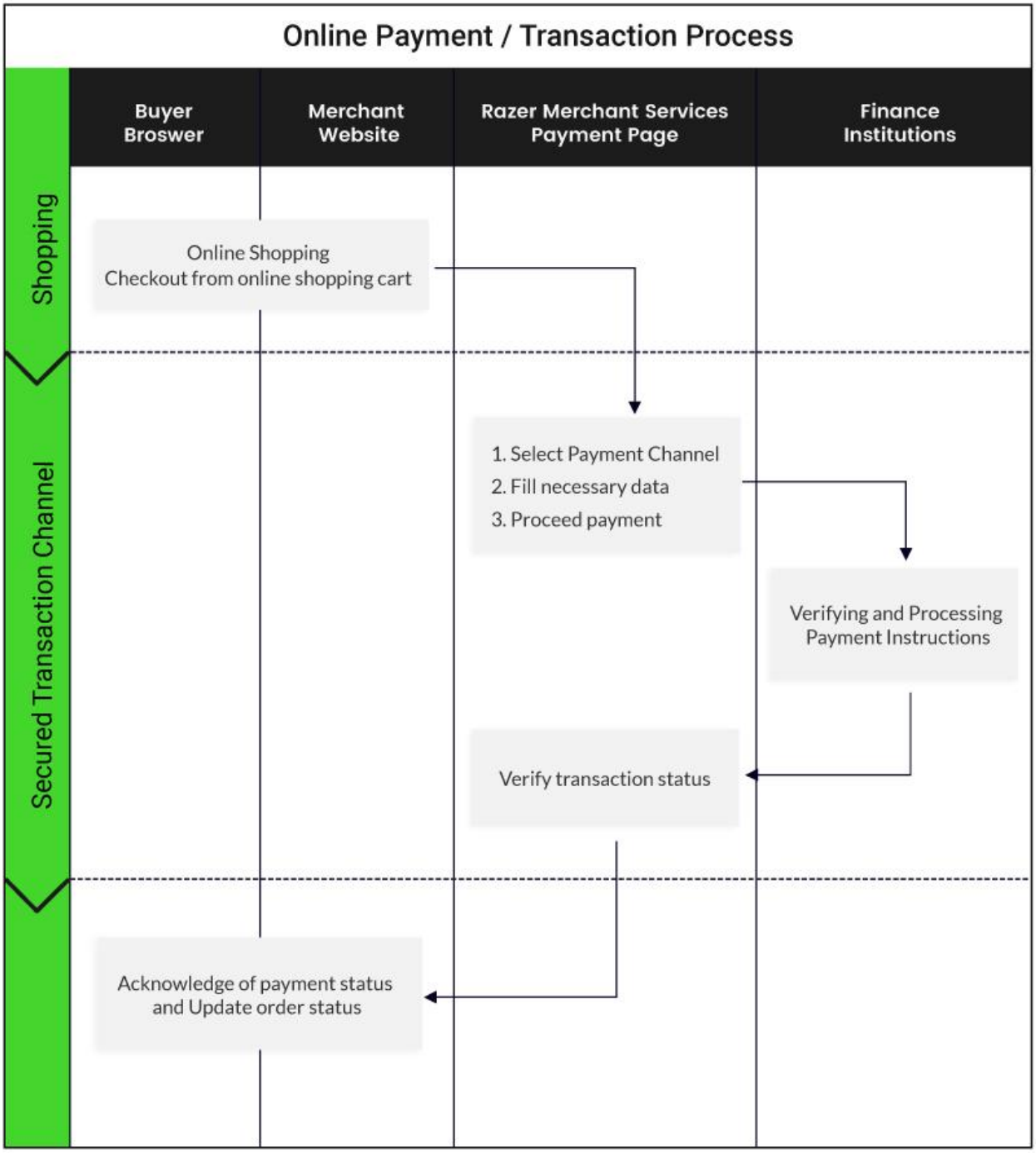
- Back-end
    - Server-to-server Notification to ensure no missing status update
    - Callback to update merchant system on deferred status change
    - Merchant can login to control panel to track payment status
    - Scheduled report on daily/weekly/monthly basis to update merchant via email
    - Real-time visualized reports

PAYMENT FLOW OVERVIEW

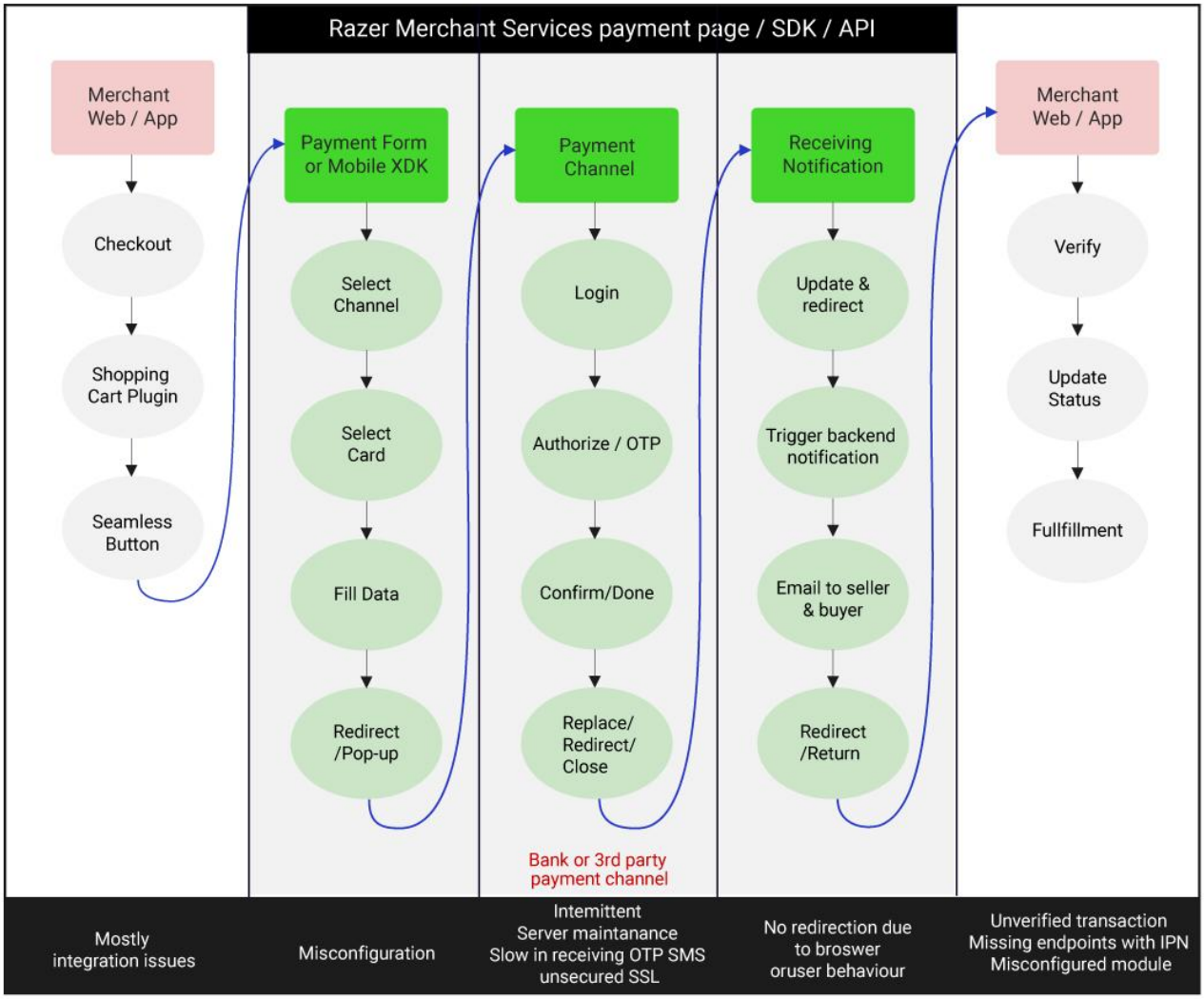
RMS provides hosted payment page service, so that merchants do not require costly and tech-savvy PCI-DSS compliance at merchant websites or systems. The integration is as simple as passing parameters via HTTPS **POST** or **GET** method from merchant to RMS payment page. Buyer will proceed their transaction on internet banking or any payment channel. Once completed, RMS will redirect the buyer's front-end back to the merchant system, using the POST method.

IPN (instant payment notification) or ACK from merchants could be implemented to confirm the receiving of payment status updates.





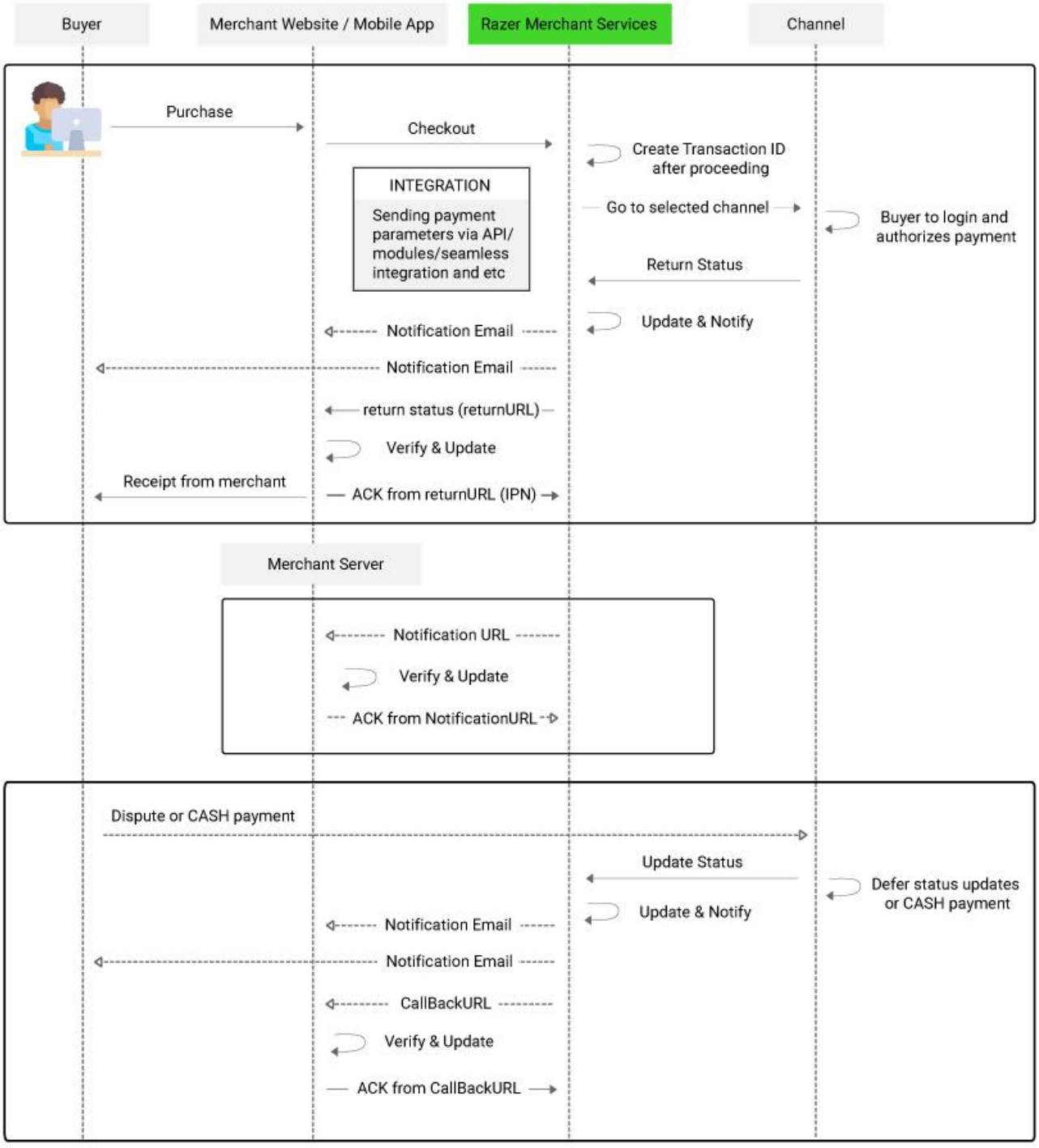
How to identify and report tech-related issue to RMS?



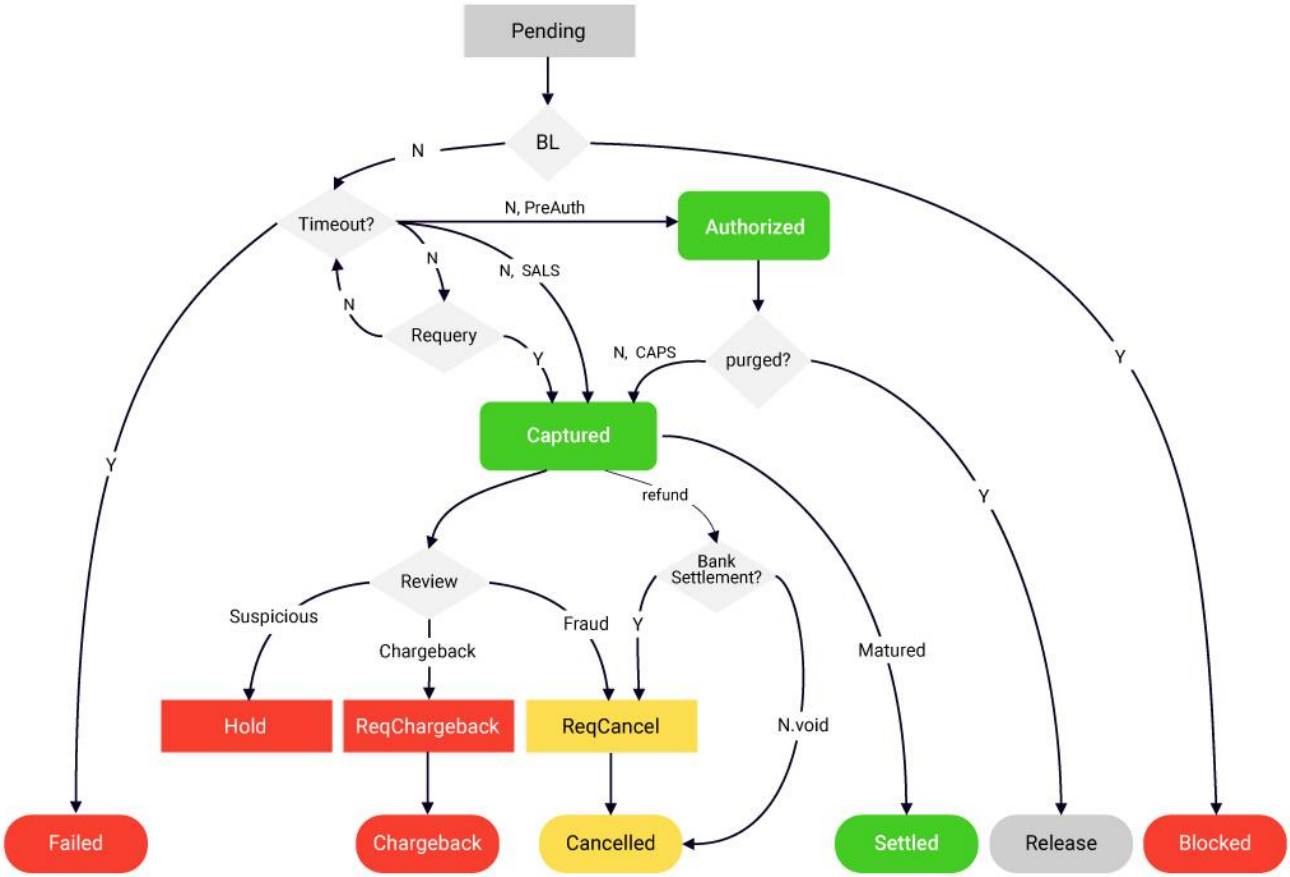
Merchant needs to prepare 3 simple and similar payment endpoint scripts to handle the payment notification from the payment gateway:

1. **return URL**: front-end or browser redirection notification, which are normally not a 100% reliable and robust channel due to unexpected network connectivity issue or client-side behavior, such as browser application crashes;
2. **notification URL**: a server-to-server back-end webhook which is more robust and crucial, especially for mobile application;
3. **callback URL**: a special handler webhook to get notified on non-realtime payment status, such as “deferred status update”, change of payment status, or Razer Cash, which is not a realtime payment naturally.

After the normal payment flow, merchants can always send payment status query requests, which is defined in **reQuery** APIs (a.k.a PSQ, Payment Status Query).



Razer Merchant Services Payment Status Flow



SECURITY & DATA INTEGRITY

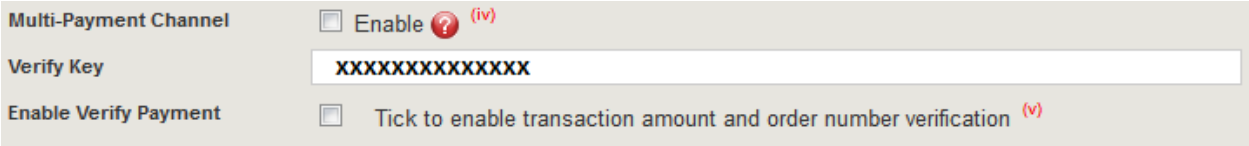
For online payment. The system uses "***merchant ID***", "***Verify Key***"(public key, a.k.a verification key) and "***Secret Key***"(private key) to generate encrypted hash string to ensure data integrity in the payment process.

Verify Key (Public Key)

Verify Key is unique shared secret for RMS merchants to generate request to the payment gateway. It is a key or seed for generating one-time hash data, which are known as "***vcode***" (merchant's payment request to the gateway) or some of the "***skey***" (APIs' request from merchant to the gateway).

How to get the verify key?

1. Logon to RMS Merchant Portal website
2. Go to the Merchant Profile tab.
3. Scroll down until you see the word "Verify Key"
4. Get the value and use it on any functions that require it.



xxxxxxxxxxxxxx is Merchant's Verify Key provided by the payment gateway. Please make sure it is at least 32 characters. Merchant may request to change the key when necessary.

Secret Key (Private Key)

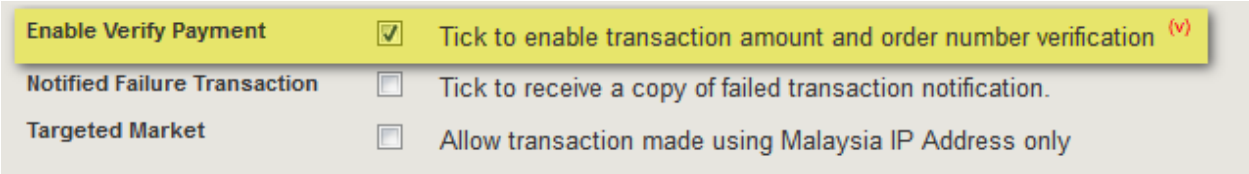
Secret Key is a unique shared secret for the payment gateway to generate responses to merchants. It is a key or seed for generating one-time hash data, which are known as "***skey***" (in the payment response) or some of the "***VrfKey***" (APIs' response from payment gateway). Merchant or developer is advised **NOT** to disclose this secret key to the public. Once the key is suspected to be compromised, please contact RMS customer service immediately to reset the key.

Merchants who are using RMS mobile xdk MUST use different strings for "*Verify Key*"(public key) and "*Secret Key*"(private key), respectively. Contact RMS customer service to reset your keys whenever needed.

vcode (in payment request)

vcode is to ensure the data integrity passed from merchant-end (either website or mobile APP) to the payment page to avoid man-in-the-middle (MITM) attack. It uses "*Verify Key*"(public key) in combination with the data string for hashing purposes.

It becomes mandatory for each transaction if "Enable Verify Payment" is activated in merchant profile as shown:-



vcode was encrypted using MD5 encryption hash function and consists of the following information (must be set in the following orders) :

1. Transaction amount
2. Merchant ID
3. Order ID
4. Verify Key

Formula to generate *vcode*

vcode = md5(amount & merchantID & orderID & verify_key)

Example to generate *vcode* for PHP developer

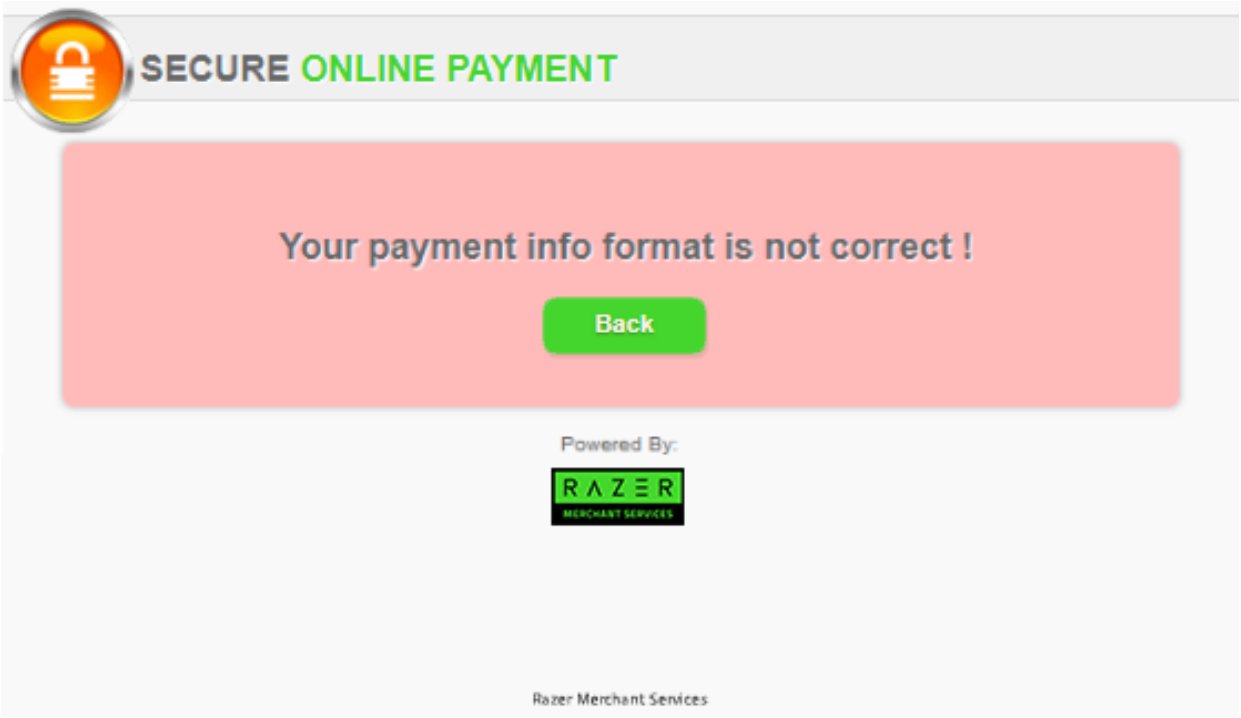
```
<?php

$amount      = "27.60";
$merchantID = "ACME";
$orderid    = "OD8842";
$verifykey = "xxxxxxxxxxxxxxxxxx";
// Replace xxxxxxxxxxxxxxxxxx with your Verify Key

// vcode formula
$vcode       = md5( $amount.$merchantID.$orderid.$verifykey );

// output of the vcode based on above information equals to :
$vcode       = "ec7f2c6e85769728a5e9b75893ee6bc1";

?>
```

Verification tool for *vcode*

To verify whether the ***vcode*** generated is correct, merchant may check on this URL:-
https://api.merchant.razer.com/RMS/query/vcode.php

What happens if a merchant passes in an incorrect ***vcode***?**

An error will be displayed on the payment page as shown:-



skey (in payment response)

skey is a payment gateway generated returning hash string to ensure the payment result data integrity that passed to the merchant system. Merchants MUST verify this hash string properly and compare the order ID, currency, amount, and also the payment date/time, to protect self interest from being cheated by a fraudster/hacker. It uses a "*Secret Key*"(private key) in combination with a data string for the hashing process.

Please note that there are other "***skey***" with different formulas in **Merchant Request APIs**' parameters. Try not to confuse yourself with this payment response "***skey***".

skey was encrypted twice using MD5 encryption hash function and consists of the following information (must be set in the following orders) :

First hash string

1. Transaction ID
2. Order ID
3. Status
4. Merchant ID (domain)
5. Amount
6. Currency

Final hash string

1. Payment Date/Time
2. Merchant ID (domain)
3. First hash string
4. Approval Code
5. Secret Key

Formula to generate *skey*

pre_skey = md5(txnID & orderID & status & merchantID & amount & currency)
skey = md5(paydate & merchantID & *pre_skey* & appcode & secret_key)

Example to generate *skey* for PHP developer

```
<?php

$vkey ="xxxxxxxxxx"; //Replace xxxxxxxxxx with your Secret_Key

/*********************************
*Don't change below parameters
*********************************/
$tranID =       $_POST['tranID'];
$orderid        =       $_POST['orderid'];
$status         =       $_POST['status'];
$merchant       =       $_POST['domain'];
$amount         =       $_POST['amount'];
```

```
$currency       =       $_POST['currency'];
$appcode        =       $_POST['appcode'];
$paydate        =       $_POST['paydate'];
$skey   =       $_POST['skey']; //Security hashstring returned by PG

/*************************************************************
* To verify the data integrity sending by PG
*************************************************************/
$key0 = md5( $tranID.$orderid.$status.$merchant.$amount.$currency );
$key1 = md5( $paydate.$merchant.$key0.$appcode.$vkey );
//key1 : Hashstring generated on Merchant system
// either $merchant or $domain could be one from POST
// and one that predefined internally
// by right both values should be identical

if( $skey === $key1 ){
  // If matched, perform another extra checking before approved order

} elseif( $skey != $key1 ){
  // If unmatched, reject the order or merchant might send query to
  // PG using Merchant requery to double check payment status
  // for that particular order.
} else {
  // error or exception case
}

?>
```

SIMULATION

Merchants should be provided a sandbox account or developer account from RMS, in order to simulate the payment flow for successful and failure transaction cases.
Please note that the sandbox account is for **SIT** purposes and only simulated credit card channel is available. **Sandbox** environment is solely for basic integration and validation purposes, **DO NOT** treat it as a grant for production environment testing and the configuration and implementation might be different. "**De**v" account has a full range of production features for advanced integration and **UAT** purposes.

Sandbox Account

- Sandbox provides only simple simulation on card payment for merchant to test successful and failure cases based on algorithm (Luhn-10) and no actual transaction occurs between bank or payment gateway
- Many advanced features are not available in the sandbox environment
- Test card - sample for Visa and Mastercard card number for testing:

Pattern	MasterCard	Visa
Positive Test	5105105105105100 (CVV: 444) 5555555555554444 (CVV: 444)	4111111111111111 (CVV: 111) 4012888888881881 (CVV: 111)
Negative Test	5555555555554440 (CVV: 444)	4111111111111110 (CVV: 111)

- Expiry date for credit card need to be valid (NOT expired) as of the current month and year.

Developer Account

For full feature integration and **UAT**, such as mobile xdk or seamless integration, please request "Developer" account or "Dev" account from RMS. These accounts are all set with a short period of lifespan for integration and testing purposes only. No settlement and mass payout is allowed in these accounts.
For real transactions conducted in a "Dev" account, please void or cancel the transaction on the same day. For Malaysian internet banking, only Maybank, CIMB Clicks, and FPX offers refund feature. You may request to turn on the channel if it is not enabled in your "Dev" account.
A "Dev" or developer account is always ended in "**_Dev**" and there will be **no settlement** (hence no payout) for this type of account. Please make sure that your "Dev" account has the same configuration in the production merchant profile to guarantee the expected test results.

PAYMENT APIs

Initiating Payment

Passing parameters to the payment gateway hosted payment page using POST/GET method via HTTPS or SSL connection will initiate a payment request from the merchant system.

REQUEST URL

Production: https://pay.merchant.razer.com/RMS/pay/***MerchantID*/*Payment_Method***

Sandbox: https://sandbox.merchant.razer.com/RMS/pay/***MerchantID*/*Payment_Method***

MerchantID: mandatory value to identify merchant;

Payment_Method: optional value to identify payment channel; Payment channel can be identified by using ***channel filename***, or pass the ***channel parameter*** to **index.php** using GET/POST method.

The URL is an API to accept POST/GET parameters from the merchant site as well as the payment page for buyers. For merchants who use the POST method please include the " / " after ***MerchantID*** on the URL.

WARNING: Loading the bank login page or authorization page into iframe is not recommended as most of the banks or channels will block all cross-origin requests and treat it as an XFS, XSS, and CSRF attack.

Channel Lists

Multi-Currency Payment

Default currency: -

Channel Name	*filename*	*channel*	Payable Amount Range	Extra Information
Visa / MasterCard	indexT.php	creditT	> 1.00	51 currencies supported : AED, **AUD**, **BND**, BRL, CAD, CHF, **CNY**, COP, CZK, DKK, EGP, **EUR**, FJD, **GBP**, **HKD**, HUF, **IDR**, ILS, INR, IQD, IRR, ISK, **JPY**, **KRW**, KWD, LAK, LKR, MOP, MXN, **MYR**, NGN, NOK, NZD, **PHP**, PKR, PLN, QAR, RUB, SAR, SCR, SEK, **SGD**, **THB**, TRY, **TWD**, **USD**, **VND**, ZAR, BDT, NPR, MMK
Visa / MasterCard	indexW.php	creditW	> 1.00	THB, USD

Malaysia

Default currency: MYR

Channel Name	*channel filename*	*channel parameter*	Payable Amount Range / Downtime*	Extra Information
Visa / MasterCard	index.php	credit	> 1.00	credit/debit/prepaid card
Visa / MasterCard	indexAB.php	creditAB	> 1.00	credit/debit/prepaid card
FPX B2C				
FPX B2C Bank Islam	BIMB.php	BIMB	> 1.00	indirect internet banking
FPX B2C Maybank	MB2U.php	MB2U	> 1.00 / 12am - 12:15am	indirect internet banking
FPX B2C Public Bank	PBB.php	PBB	> 1.00	indirect internet banking
FPX B2C CIMBClicks	CIMBCLICKS.php	CIMBCLICKS	> 1.00	indirect internet banking
FPX B2C AmOnline	AMOnline.php	AMOnline	> 1.00	indirect internet banking
FPX B2C Hong Leong	HLBConnect.php	HLBConnect	> 1.00	indirect internet banking
FPX B2C RHB Bank	RHBNow.php	RHBNow	> 1.00 / 12am - 12:10am	indirect internet banking
FPX B2C OCBC	FPX_OCBC.php	FPX_OCBC	> 1.00	indirect internet banking
FPX B2C Standard Chartered	FPX_SCB.php	FPX_SCB	> 1.00	indirect internet banking
FPX B2C Affin Bank	FPX_ABB.php	FPX_ABB	> 1.00	indirect internet banking

FPX B2C Bank Rakyat	bankrakyat.php	bankrakyat	> 1.00 / 12am - 12:30am	indirect internet banking
FPX B2C Bank Muamalat	bankmuamalat.php	bankmuamalat	> 1.00	indirect internet banking
FPX B2C Kuwait Finance House	FPX_KFH.php	FPX_KFH	> 1.00	indirect internet banking
FPX B2C Bank Simpanan Nasional	FPX_BSN.php	FPX_BSN	> 1.00 / 12am - 12:15am	indirect internet banking
FPX B2C Alliance Bank	FPX_ABMB.php	FPX_ABMB	> 1.00	indirect internet banking
FPX B2C United Overseas Bank	FPX_UOB.php	FPX_UOB	> 1.00	indirect internet banking
FPX B2C HSBC Bank	FPX_HSBC.php	FPX_HSBC	> 1.00	indirect internet banking
FPX B2C Agrobank	FPX_AGROBANK.php	FPX_AGROBANK	> 1.00	Indirect internet banking
FPX B2B				
FPX B2B AmBank	FPX_B2B_AMB.php	FPX_B2B_AMB	2.00 - 1mil / 12am - 2am	indirect internet banking
FPX B2B Hong Leong	FPX_B2B_HLB.php	FPX_B2B_HLB	2.00 - 1mil	indirect internet banking
FPX B2B United Overseas Bank	FPX_B2B_UOB.php	FPX_B2B_UOB	2.00 - 1mil	indirect internet banking
FPX B2B Affin Bank	FPX_B2B_ABB.php .php	FPX_B2B_ABB	2.00 - 1mil	indirect internet banking
FPX B2B CIMB Bank	FPX_B2B_CIMB.php	FPX_B2B_CIMB	2.00 - 1mil	indirect internet banking
FPX B2B Bank Islam	FPX_B2B_BIMB.php	FPX_B2B_BIMB	2.00 - 1mil / 11pm - 7am	indirect internet banking
FPX B2B Public Bank	FPX_B2B_PBB.php	FPX_B2B_PBB	2.00 - 1mil	indirect internet banking
FPX B2B RHB Bank	FPX_B2B_RHB.php	FPX_B2B_RHB	2.00 - 1mil / 12am - 12:15am	indirect internet banking
FPX B2B HSBC	FPX_B2B_HSBC.php	FPX_B2B_HSBC	2.00 - 1mil	indirect internet banking
FPX B2B Maybank2E	FPX_M2E.php	FPX_M2E	2.00 - 1mil / 12am - 1 am	indirect internet banking
FPX B2B BNP Paribas	FPX_B2B_BNP.php	FPX_B2B_BNP	2.00 - 1mil / 5pm - 8.45am	indirect internet banking
FPX B2B Affin Max	FPX_B2B_ABBM.php	FPX_B2B_ABBM	2.00 - 1mil	indirect internet banking
FPX B2B Citibank	FPX_B2B_CITIBANK.php	FPX_B2B_CITIBANK	2.00 - 1mil	indirect internet banking

FPX B2B i-bizRAKYAT	FPX_B2B_BKRM.php	FPX_B2B_BKRM	2.00 - 1mil	indirect internet banking
FPX B2B Public Bank Enterprise	FPX_B2B_PBBE.php	FPX_B2B_PBBE	2.00 - 1mil	indirect internet banking
FPX B2B UOB Regional	FPX_B2B_UOBR.php	FPX_B2B_UOBR	2.00 - 1mil	indirect internet banking
FPX B2C DirectDebit e-Mandate	FPX_EMANDATE.php	FPX_EMANDATE	> 1.00 Maximum authorized installment amount but always debit MYR 1.00 upon enrollment and this will be fully refunded	direct debit enrollment
FPX B2B Agrobank	FPX_B2B_AGROBANK.php	FPX_B2B_AGROBANK	2.00- 1mil	Indirect internet banking
DuitNow				
DuitNow QR	RPP_DuitNowQR.php	RPP_DuitNowQR	> 1.00	DuitNow QR payment
Wallet				
Razer Pay	RazerPay.php	RazerPay	> 1.00	e-wallet
Razer Gold	MOLPoints.php	MOLPoints	> 1.00	e-wallet
Touch N' Go E-WALLET	TNG-EWALLET.php	TNG-EWALLET	> 1.00	e-wallet
BOOST	BOOST.php	BOOST	> 1.00	e-wallet
GrabPay	GrabPay.php	GrabPay	> 1.00	e-wallet
Maybank QRPay	MB2U_QRPay-Push.php	MB2U_QRPay-Push	> 1.00	e-wallet
ShopeePay	ShopeePay.php	ShopeePay	> 1.00	e-wallet
OTC				
Razer Cash via 7E	cash.php	cash	> 1.00	OTC @ 7-Eleven
JomPay ATM 9282	jompay.php	jompay	> 1.00	ATM bill payment
CIMB Virtual Account	CIMB-VA.php	CIMB-VA	> 1.00	CIMB Clicks / ATM transfer for CIMB customers only
Razer Cash via 99	cash99.php	cash99	> 1.00	OTC @ 99 Speedmart
Razer Cash via KK	cashkkmart.php	cashkkmart	> 1.00	OTC @ KK Mart
Installment for card				
MBB-eBPG-(PD or PW) EzyPay	MBB-EzyPay.php	MBB-EzyPay	Non-installment > 1.00 Installment: 3 mon: 200 - 1000 6 mon: 500 - 15,000	MBB card installment V/M card payment

			12 mon: 500 - 15,000 24 mon: 2,500 - 15,000	
PBB-ZIIP	installmentPBB-ZIIP.php	PBB-ZIIP	non-installment > 1.00 installment > 500.00	PBB card installment V/M card payment
CIMB-eBPG	indexL.php	creditL	non-installment > 1.00 installment > 500.00	CIMB card installment V/M card payment
AMBANK-eBP-PD	indexU.php	creditU	non-installment > 1.00 installment > 500.00	AMB card installment V/M card payment
AMBANK-eBP-PW	AMBANK-eBPG-PW.php	AMBANK-eBPG-PW	non-installment > 1.00 installment > 500.00	AMB card installment V/M card payment
HSBC-MIGS-PW	HSBC-MIGS-PW.php	HSBC-MIGS-PW	> 1.00	HSBC card installment V/M card payment
SCB-MGIS-PW	SCB-MGIS-PW.php	SCB-MGIS-PW	> 1.00	SCB card installment V/M card payment
Buy Now Pay Later				
Rely-PW	Rely-PW.php	Rely-PW	Installment 40 - 4,000	V/M card payment

* Downtime is for reference only

Remarks

1. Secure 1-click or credit card tokenization function can be enabled in the merchant profile to ease next purchase on the same merchant system.
2. Bcard loyalty reward points for successful transactions can be activated in the merchant profile. (Obsoleted)
3. For certain Razer Cash channels, such as 7-eleven and epay, round to the nearest 0 or 5 cents and convenience store fee applied. Buyer will see the “Payable Amount”, but RMS still returns the amount passed from the merchant system.

Singapore

Default currency: SGD

Channel Name	filename	channel	Payable Amount Range	Extra Information
eNETS Debit	enetsD.php	enetsD	> 5.00	indirect internet banking
SAM by SingPost	singpost.php	singpost	> 5.00	bank card / ATM
AXS	AXS.php	AXS	> 5.00	bank card / ATM
Wallet				
Razer Pay	RazerPay.php	RazerPay	> 1.00	e-wallet
PayNow	PayNow.php	PayNow	>1.00	Online banking/e-wallet

Indonesia

Default currency: IDR

Channel Name	filename	channel	Payable Amount Range	Extra Information
Artajasa	Artajasa.php	Artajasa	> 10,000.00	OTC
ATM Network Transfer Virtual Account (VA)	ATMVA.php	ATMVA	> 10,000.00 <= 10,000,000.00	cardless ATM

China

Default currency: USD/MYR/SGD (for merchant), RMB/CNY/MYR/SGD/USD (for buyer)

Channel Name	filename	channel	Payable Amount Range (Processing Currency)	Extra Information
Alipay 支付宝	alipay.php	alipay	> MYR1.00 (MYR/USD/SGD/RMB/CNY)	e-wallet (CN)
UnionPay 银联	GUPOP.php	GUPOP	> MYR1.00 (MYR/USD/CNY/AUD/CAD/EUR/ GBP/HKD/IDR/JPY/NZD/PHP/ SGD/THB/TWD/VND)	bank card
WeChat Pay 微信支付	WeChatPay.php	WeChatPay	> MYR1.00 (USD)	e-wallet (CN)

Remarks

1. Due to China Central Bank regulations, CNY(or RMB) is not able to be settled to an overseas bank account, these channels will use USD as default settlement currency. Besides USD, RMS can also settle MYR, SGD via Alipay as well and RMS will convert it to equivalent amount of settlement currency based on Maybank daily FX rate.

Thailand

Default currency: THB

Channel Name	filename	*channel*	Payable Amount Range	Extra Information
Bank of Ayudhya (Krungsri)	BAY_IB_U.php	BAY_IB_U	> 1.00	direct internet banking (fees on buyer)
Bangkok Bank	BBL_IB_U.php	BBL_IB_U	> 1.00	direct internet banking (fees on buyer)
Krung Thai Bank	KTB_IB_U.php	KTB_IB_U	> 1.00	direct internet banking (fees on buyer)
Tesco Lotus & Big Central	Cash-TH.php	Cash-TH	> 1.00	OTC
Kasikornbank PAYPLUS	KBANK_PayPlus.php	KBANK_PayPlus	> 1.00	direct internet banking (fees on buyer)
Siam Commercial Bank	SCB_IB_U.php	SCB_IB_U	> 1.00	direct internet banking (fees on buyer)
Wallet				
Kbank Promptpay QR	KBANK_THQR_Payment.php	KBANK_THQR_Payment	> 1.00	Direct internet banking & e-wallet

Vietnam

Default currency: VND

Channel Name	filename	*channel*	Payable Amount Range	Extra Information
NganLuong	nganluong.php	nganluong	> 2,000.00	e-wallet
VTC Pay	vtcpay.php	vtcpay	> 1,000.00	e-wallet

Philippines

Default currency: PHP

Channel Name	filename	*channel*	Payable Amount Range	Extra Information
Dragonpay	dragonpay.php	dragonpay	> 50.00	indirect internet banking / cash
Razer Cash PH	cashph.php	cashph	> 100	OTC @ Razer Cash PH

Taiwan

Default currency: TWD

Channel Name	filename	*channel*	Payable Amount Range	Extra Information
ESUN Cash-711	ESUN_Cash711.php	ESUN_Cash711	>= 1.00	OTC
ESUN Cash FamilyMart	ESUN_CashFamilyMart.php	ESUN_CashFamilyMart	>= 1.00	OTC
ESUN ATM	ESUN_ATM.php	ESUN_ATM	>= 1.00	Virtual Account
ESUN Cash Hi-Life	ESUN_CashHiLife.php	ESUN_CashHiLife	>= 1.00	OTC

Payment Page Integration

This is the traditional integration method which will send the buyer information to the payment gateway hosted payment page. You may want to explore *seamless integration*, *Direct Server API*, *mobile xdk* or supported shopping cart payment modules for better user experience beside this hosted payment page integration on github repositories. Recurring API is also available for card payment and DirectDebit.

Request Parameters

These parameters can be passed using either POST or GET method, or mixed of these methods. Please use UTF-8 encoding for all values.

Variable / Parameter	Type Format / Max Length	Description / Example
merchant_id	optional, alphanumeric	Merchant ID provided and it is on the request URL, otherwise need to pass this as parameter
amount	mandatory, integer or up to **2** decimal points numeric value	The total amount to be paid in one purchase order. Configurable to lock this field (Read-only). E.g. 500, 168.99, comma(,) is not allowed
orderid	mandatory, alphanumeric up to 32 characters	Invoice or order number from merchant system. Can be set to Read-only field. E.g. BH2018-09rev
bill_name	mandatory, alphanumeric, 128 chars	Buyer's full name.
bill_email	mandatory, email, 128 chars	Buyer's email address. Must be valid email.
bill_mobile	mandatory, alphanumeric, 32 chars	Buyer's mobile number or contact number. Must be valid mobile number.
bill_desc	mandatory, alphanumeric, 64kB	Purchase itemized list or order description. Try to avoid special characters so that the payment request is not blocked by the web application firewall. For Taiwan channels please refer to **Guideline for Taiwan channels**.
b_addr1	optional, mandatory for specific channel	Billing address line 1
b_addr2	optional, mandatory for specific channel	Billing address line 2
b_zipcode	optional, mandatory for specific channel	Billing address zipcode or postcode
b_city	optional, mandatory for specific channel	Billing address city
b_state	optional, mandatory for specific channel	Billing address state
country	mandatory, 2 chars of ISO-3166 country code (Alpha-2)	Buyer's country E.g. MY for Malaysia.
vcode	conditional if accept open amount or order payment, such as virtual terminal is optional otherwise is mandatory. 32 chars hexadecimal string	This is the data integrity protection hash string. Refer ***vcode*** section for details.

cur / currency	optional, 3 chars ISO-4217 currency code	Default payment currency from merchant site. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR
channel	optional, predefined string. Refer to the channel column in the Channel Lists.	Default payment page will be displayed without *channel* specified.
langcode	optional, predefined string. Currently only *en* for English & *cn* for Simplified Chinese.	Default language, i.e. English, will be displayed without the langcode specified.
returnurl	optional, URL	For selected merchants only. Used for multiple return URLs. All URLs must be registered beforehand with RMS. This will slow down your payment request.
callbackurl	optional, URL	Used for both multiple callback URL and notification URL. All URL endpoints must be registered beforehand with RMS.
cancelurl	optional, URL	Buyer will redirect to this URL if the buyer clicks the "Cancel" button to abandon payment before proceeding. No transaction will be created.
s_name	optional, mandatory for specific channel	Receiver name
s_addr1	optional, mandatory for specific channel	Delivery address line 1
s_addr2	optional, mandatory for specific channel	Delivery address line 2
s_zipcode	optional, mandatory for specific channel	Delivery destination zipcode
s_city	optional, mandatory for specific channel	Delivery destination city
s_state	optional, mandatory for specific channel	Delivery destination state
s_country	optional, mandatory for specific channel	Delivery destination country
s_merchantID	conditional for partner	Partner's sub-merchant MID
s_merchantURL	conditional for partner	Partner's sub-merchant URL
s_merchantMCC	conditional for partner	Partner's sub-merchant MCC
is_escrow	optional, for escrow payment	Set the value to 1 to indicate this is an escrow payment. Escrow transactions also can be set after the transaction has been created using escrow API.
non_3DS	conditional for partner, 0 or 1	Applicable to card processing via specific processor using specific currency for pre-approved partners only. Equal to 0 by default and 1 for non-3DS transaction
tcctype	optional, for card type transaction	Available value is: SALS = Capture Transaction (Default) AUTH = Authorize Transaction (Please inform RMS before starting using pre-auth)
token	optional	Token value for 1-click payment to show the selected card
installmonth	optional, mandatory for installment payment	Total month of installment. E.g: 0, 3, 6, 12, 24

cash_waittime	optional, integer	To overwrite the allowed waiting time for cash payment in hour(s), values greater than "Payment Expiry Time" in merchant profile will be capped.
split_info	optional, alphanumeric with pipe (|) as data delimiter and comma (,) as recipient separator	Used for "Alipay Split Payment". Format as below without '**<**' and '**>**':- <submer_ID>|<amount>, <submer_ID>|<amount>
AppDeeplink	optional, URL	Mobile deeplink/universal link that allows e-wallet In-App payment.

Examples

Using PHP to generate a GET payment request

```
<?PHP
echo "<a href=https://pay.merchant.razer.com/RMS/pay/merchantID/?";
echo "amount=".$amount."&";
echo "orderid=".urlencode($oid)."&";
echo "bill_name=".urlencode($name)."&";
echo "bill_email=".urlencode($email)."&";
echo "bill_mobile=".urlencode($mobile)."&";
echo "bill_desc=".urlencode($description)."&";
echo "country=".$country."&";
echo "vcode=".$vcode."> Pay Now </a>";
?>
```

Using PHP to generate a POST form payment request

```
<?PHP
echo "<form action='https://pay.merchant.razer.com/RMS/pay/merchantID/' method=POST >";
echo "<input type=hidden      name=amount              value='$amount'>";
echo "<input type=hidden      name=orderid         value='$oid'>";
echo "<input type=hidden      name=bill_name     value='$name'>";
echo "<input type=hidden      name=bill_email    value='$email'>";
echo "<input type=hidden      name=bill_mobile   value='$mobile'>;
echo "<input type=hidden      name=bill_desc     value='$description'>";
echo "<input type=hidden      name=country         value='$country'>";
echo "<input type=hidden      name=vcode               value='$vcode'>";
echo "<input type=submit              value=' PAY NOW '>";
echo "</form>";
?>
```

Using ASP to generate a GET payment request

```
<%
```

```
dim amount, orderid, name, email, mobile, desc, country, vcode
amount    = ”18.99”
orderid= ”DG873MH370”
name       = ”Mr Albert Anderson”
email       = ”a.anderson@somewhere.com”
mobile = ”016-2341234”
desc        = ”DIGI Reload Coupon RM20 with discount”
country= ”MY”
vcode      = md5(amount & merchantID & orderid & xxxxxxxxxxxx )
// REPLACE xxxxxxxxxxxx with Verify Key
response.write("<a href=https://pay.merchant.razer.com/RMS/pay/merchantID/?")
response.write("amount=” & amount)
response.write("&orderid=” & orderid)
response.write("&bill_name=” & name)
response.write("&bill_email=” & email)
response.write("&bill_mobile=” & mobile)
response.write("&bill_desc=” & desc)
response.write("&country=” & country)
response.write("&vcode=” & vcode)
response.write(”’> Pay Now </a>”)
%>
```

Using ASP to generate a POST form payment request

```
<%
dim amount, orderid, name, email, mobile, desc, country, vcode
amount    = ”28.99”
orderid= ”DG873MH37o”
name       = ”Mr Samuel Lim”
email       = ”sam.lim@nowhere.com”
mobile     = ”86-232389872”
desc        = ”Facebook Prepaid Reload Coupon RM30 with discount”
country= ”CN”
vcode      = md5(amount & merchantID & orderid & xxxxxxxxxxxx )
// REPLACE xxxxxxxxxxxx with Verify Key
response.write(“<form action=’https://pay.merchant.razer.com/RMS/pay/merchantID/’ method=POST >)
response.write(“<input type=hidden name=amount value=”&amount&”>)
response.write(“<input type=hidden name=orderid value=”&orderid& ”>)
response.write(“<input type=hidden name=bill_name value=”&name& ”>)
response.write(“<input type=hidden name=bill_email value=”&email&”>)
response.write(“<input type=hidden name=bill_mobile value=”&mobile&”>)
response.write(“<input type=hidden name=bill_desc value=”&desc&”>)
response.write(“<input type=hidden name=country value=”&country&”>)
response.write(“<input type=hidden name=vcode value=”&vcode&”>)
response.write(“<input type=submit value=’ PAY NOW ’ ”>)
response.write(“</form>”)
%>
```

Getting Payment Result

Payment results will be returned to the merchant system once payment is done or the user abandons the payment process. HTTP POST is the only method that payment gateway returns all parameters to a merchant's return URL for real-time status update, which the merchant can configure in merchant admin. Merchant system should block all other methods or parameters from an untrusted source.

Response Parameters

Variable / Parameter	Type Format / Max Length	Description / Example
amount	2 decimal points numeric value	The total amount paid or to be paid in Razer Cash payment request.
orderid	alphanumeric, 32 characters	Invoice or order number from merchant system.
tranID	integer, 10 digits	Unique transaction ID for tracking purpose.
domain	alphanumeric, 32 chars	Merchant ID in RMS PG system.
status	2-digit numeric value	00 for Successful payment, 11 for failed status, 22 if pending.
appcode	alphanumeric, 16 chars	Bank approval code. Mandatory for card payment. Certain channels return empty value.
error_code	alphanumeric	Refer to the Error Codes section.
error_desc	text	Error message or description.
skey	32 chars hexadecimal string	This is the data integrity protection hash string. Refer ***skey*** section for details.
currency	2 or 3 chars (ISO-4217) currency code	Default currency is MYR (indicating Malaysia Ringgit) for Malaysia channels.
channel	predefined string in PG system	Channel references for the merchant system. Refer to the below table.
paydate	Date/Time(YYYY-MM-DD HH:mm:ss)	Date/Time of the transaction.
extraP*	optional (on request)	JSON encoded string or array token: 16-digit card token for merchant to store for recurring MIT (merchant initiated transaction) fraudscreen: 1-digit integer, i.e. 1=Unknown, 2=Passed, 3=Alert, 4=Suspicious, 5=Fraud fpx_txn_id: FPX transaction ID fpx_buyer_name: FPX buyer name ccbrand: Visa, MasterCard, AMEX bank_issuer: The issuing bank or institution ccbin: Issuer/Bank identification number, first 6-digit

		cclast4: Last 4-digit of card number cctype: Credit, Debit, Prepaid

Value of "*channel*" in return URL

Card / Wallet		
Malaysia & International		
Credit/Debit/Prepaid card (Visa/MasterCard) Credit **Razer Gold - Pay with Razer (PWR)** MOLPoints **WeChat Pay MY (Online)** WeChatPayMY **Maybank QRPay** MB2U_QRPay-Push **DuitNow QR** RPP_DuitNowQR	**Point BCard** Point-BCard **Webcash** WEBCASH **WeChat Pay MY (Offline)** WeChatPayMY-Offline **Touch N' Go E-WALLET** TNG-EWALLET	**PayPal** PayPal **Boost** BOOST **GrabPay** GrabPay **ShopeePay** ShopeePay
China		
Alipay (Online) Alipay	**WeChat Pay (Online)** WeChatPay	**WeChat Pay (Offline)** WeChatPay-Offline
Thailand		
Paysbuy paysbuy	**Kbank Promptpay QR** KBANK_THQR_PAYMENT	
Vietnam		
Nganluong NGANLUONG **VTC Pay eWallet & Banks** VTCP_EWB	**VTC-Pay** VTC-Pay **VTC Pay eWallet & Credit Cards** VTCP_EWC	**VTC Pay eWallet** VTCP_EW
Indirect / Direct Internet Banking		
Malaysia		
FPX FPX **RHB Now** FPX_RHB / RHB-ONL **Bank Islam** FPX_BIMB **Affin Online** FPX_ABB / Affin-EPG **myBSN**	**Maybank2u** FPX_MB2U / MB2u **PBe** FPX_PBB / PBeBank **AmOnline** FPX_AMB / AMB-W2W **i-Muamalat** FPX_BMMB **OCBC Online**	**CIMB Clicks** FPX_CIMBCLICKS / CIMB-Clicks **Hong Leong Connect / ~~PEx+~~** FPX_HLB / HLB-ONL / ~~PEXPLUS~~ **Alliance online** FPX_ABMB / ALB-ONL **i-Rakyat** FPX_BKRM **UOB Online**

FPX_BSN	FPX_OCBC	FPX_UOB
HSBC Online FPX_HSBC	**Standard Chartered Online** FPX_SCB	**KFH Online** FPX_KFH
FPX B2B Model FPX_B2B	**FPX B2B AmBank** FPX_B2B_AMB	**FPX B2B Hong Leong** FPX_B2B_HLB
FPX B2B United Overseas Bank FPX_B2B_UOB	**FPX B2B Affin Bank** FPX_B2B_ABB	**FPX B2B HSBC** FPX_B2B_HSBC
FPX B2B Maybank2E FPX_M2E	**FPX B2B CIMB Bank** FPX_B2B_CIMB	**FPX B2B Bank Islam** FPX_B2B_BIMB
FPX B2B RHB Bank FPX_B2B_RHB	**FPX B2B Public Bank** FPX_B2B_PBB	**FPX B2B Kuwait Finance House** FPX_B2B_KFH
FPX B2B Deutsche Bank FPX_B2B_DEUTSCHE	**FPX B2B Alliance Bank** FPX_B2B_ABMB	**FPX B2B Standard Chartered** FPX_B2B_SCB
FPX B2B OCBC FPX_B2B_OCBC	**FPX B2B Bank Muamalat** FPX_B2B_BMMB	**FPX e-Mandate** FPX_EMANDATE
FPX DirectDebit FPX_DIRECTDEBIT	**Agrobank** FPX_AGROBANK	**FPX B2B Agrobank** FPX_B2B_AGROBANK
FPX B2B Affin Max FPX_B2B_ABBM	**FPX B2B Citibank** FPX_B2B_CITIBANK	**FPX B2B i-bizRAKYAT** FPX_B2B_BKRM
FPX B2B Public Bank Enterprise FPX_B2B_PBBE	**FPX B2B UOB Regional** FPX_B2B_UOBR	
Singapore		
eNets Debit eNets-D	**eNets Debit DBS Bank** eNets-D_DBS	**eNets Debit DBS CITI Bank** eNets-D_CITIBank
eNets Debit United Overseas Bank eNets-D_UOB	**eNets Debit OCBC Bank** eNets-D_OCBC	**eNets Debit Standard Chartered Bank** eNets-D_SCB
Thailand		
Bangkok Bank TH_BBLPN / BBL_IB & BBL_IB_U	**Bank of Ayudhya (Krungsri)** TH_BAYPN / BAY_IB_U	**Krung Thai Bank** TH_KTBPN / KTB_IB & KTB_IB_U
Siam Commercial Bank TH_SCBPN	**OMISE** OMISE	
Vietnam		
VTC Pay Banks VTCP_B	**VTC Pay - Vietcombank** VTCP_Vietcombank	**VTC Pay - Techcombank** VTCP_Techcombank
VTC Pay - MB VTCP_MB	**VTC Pay - Vietinbank** VTCP_Vietinbank	**VTC Pay - Agribank** VTCP_Agribank
VTC Pay - DongABank VTCP_DongABank	**VTC Pay - Oceanbank** VTCP_Oceanbank	**VTC Pay - BIDV** VTCP_BIDV
VTC Pay - SHB VTCP_SHB	**VTC Pay - VIB** VTCP_VIB	**VTC Pay - MaritimeBank** VTCP_MaritimeBank
VTC Pay - Eximbank VTCP_Eximbank	**VTC Pay - ACB** VTCP_ACB	**VTC Pay - HDBank** VTCP_HDBank

VTC Pay - NamABank VTCP_NamABank **VTC Pay - VietABank** VTCP_VietABank **VTC Pay - SeaABank** VTCP_SeaABank **VTC Pay - GPBank** VTCP_GPBank **VTC Pay - ABBANK** VTCP_ABBANK	**VTC Pay - SaigonBank** VTCP_SaigonBank **VTC Pay - VPBank** VTCP_VPBank **VTC Pay - PGBank** VTCP_PGBank **VTC Pay - BACABANK** VTCP_BACABANK **VTC Pay - LienVietPostBank** VTCP_LVPB	**VTC Pay - Sacombank** VTCP_Sacombank **VTC Pay - TienPhongBank** VTCP_TienPhongBank **VTC Pay - Navibank** VTCP_Navibank **VTC Pay - PHUONGDONG** VTCP_PHUONGDONG **VTC Pay - BVB** VTCP_BVB
China		
Union Pay UPOP		
Kiosk / Over-the-Counter / Cash / Bill Payment		
Malaysia		
7-Eleven Cash-711 **Jompay** jompay	**99SpeedMart** Cash-99SM **Cash-in to PG** Cash-Deposit **CIMB Virtual Account** CIMB-VA	**Maybank ATM** Cash-MBBATM
Singapore		
SingPost ATM Cash-SAM	**AXS Kiosk** AXS	
Indonesia		
ATM Transfer via Permata Bank ATMVA	**Artajasa** Artajasa	
Philippines		
DragonPay dragonpay	**Razer Cash PH** Cash-PH	
Thailand		
Tesco Lotus Cash-TH	**Big Central** Cash-TH	
Taiwan		
FamilyMart 全家便利商店 FAMILYMART		

* Channel in yellow text means obsoleted

Merchant is strongly recommended to implement IPN(instant payment notification) in order to acknowledge(**ACK**) on the receiving of payment status from the payment gateway. There are 2 ways to implement IPN. Please refer to the IPN section for details.

Examples of Payment Endpoint

You may use the sample for all 3 endpoints, i.e. return URL, notification URL, and callback URL by making little modification based on your own requirements.

Sample return URL script for PHP

```php
<?php

$vkey ="xxxxxxxxxxx"; //Replace xxxxxxxxxxx with Secret_Key

/**********************************
*Don't change below parameters
**********************************/
$tranID    =        $_POST['tranID'];
$orderid   =        $_POST['orderid'];
$status    =        $_POST['status'];
$domain    =        $_POST['domain'];
$amount    =        $_POST['amount'];
$currency           =        $_POST['currency'];
$appcode            =        $_POST['appcode'];
$paydate   =        $_POST['paydate'];
$skey               =        $_POST['skey'];

/************************************************************
* To verify the data integrity sending by PG
************************************************************/
$key0 = md5( $tranID.$orderid.$status.$domain.$amount.$currency );

$key1 = md5( $paydate.$domain.$key0.$appcode.$vkey );


if( $skey != $key1 ) $status= -1; // Invalid transaction.
// Merchant might issue a requery to PG to double check payment status

if ( $status == "00" ) {
  if ( check_cart_amt($orderid, $amount) ) {
  /*** NOTE : this is a user-defined function which should be prepared by merchant ***/
  // action to change cart status or to accept order
  // you can also do further checking on the paydate as well
  // write your script here .....
  }
} else {
  // failure action. Write your script here .....
  // Merchant might send query to PG using Merchant requery
  // to double check payment status for that particular order.
}

// Merchant is recommended to implement IPN once received the payment status
// regardless the status to acknowledge the PG

?>
```

Sample return URL script for ASP/ASP.NET

```
<!--#include file="md5.asp"-->  'For ASP Developer
<!--#include file="md5.aspx"--> 'For ASP.NET Developer

<%
'  md5.asp/md5.aspx is a 3rd party developed md5 solution for ASP/ASP.NET user
' You could get the md5.asp/md5.aspx from support-sa@razer.com
' Some variables below are coming from POST method

dim key0, key1, tranID, orderid, status, merchantID, amount, currency, paydate, appcode, skey

tranID      = Request.Form("tranID")
orderid     = Request.Form("orderid")
status      = Request.Form("status")
merchantID          = Request.Form("domain")
amount      = Request.Form("amount")
currency    = Request.Form("currency")
paydate     = Request.Form("paydate")
appcode     = Request.Form("appcode")
skey                = Request.Form("skey")
key0                = md5( tranID & orderid & status & domain & amount & currency )
key1                = md5( paydate & merchantID & key0 & appcode & "xxxxxxxxxxx" )
'Replace xxxxxxxxxxx with Secret_Key

' invalid transaction if the key is different. Merchant might issue a requery to PG to double check payment status

If skey <> key1 then

  status= -1

End if


If status = "00" then
 '  checking the validity of cart amount & orderid.
 ' if the verification test passed then can update the order status to paid.
 ' you can also do further checking on the paydate as well
Else
 ' failure action
 ' Merchant might send query to PG using merchant requery
 ' to double check payment status for that particular order.
End if

' Merchant is to implement IPN to ack on receiving of payment status
' regardless the payment status

%>
```

Payment Status Notification (Merchant Webhook or the 3 Endpoints)

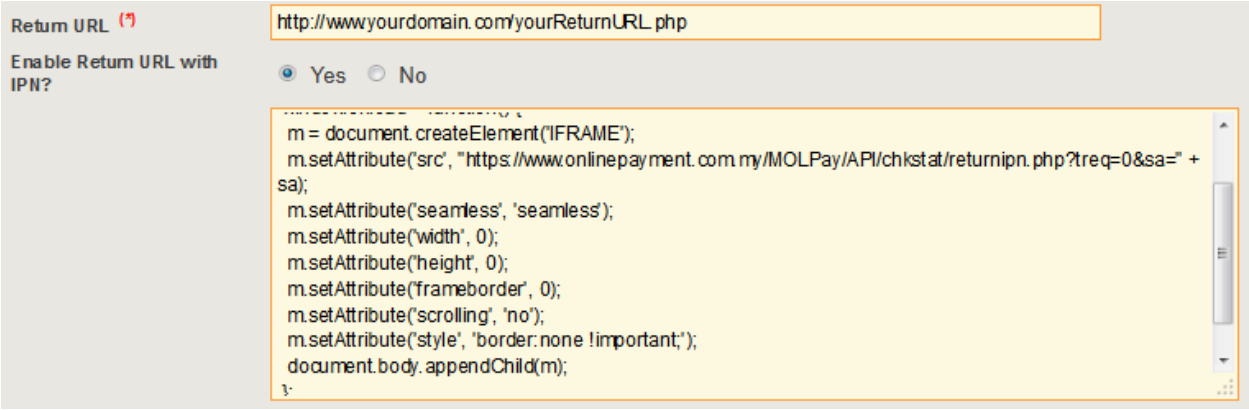
WARNING: Please note that multiple payment notifications (either from ***return URL***, ***notification URL*** or ***callback URL***) for a single transaction is possible but this does not mean that the buyer has paid twice or multiple times.

Return URL with IPN (Instant Payment Notification)

For normal payment flow, the buyer browser is being redirected to a hosted payment page, financial institution or channel page(if any), and then returned to the merchant website or system. Users might close the browser any time throughout the payment process, even if the payment is completed, successfully or failed. Another possible reason that rarely happens is the network connectivity issue. As a result, the payment gateway is unable to update the merchant system on the payment status. Therefore, merchants are recommended to implement IPN to acknowledge(ACK) upon the receiving of status from gateway. Otherwise the callback worker will resend the payment status within a time interval.

Implementation:

Step 1: Logon to merchant admin, choose “Yes” to “Enable Return URL with IPN”, as shown:-



Step 2: There are 2 approaches to ack on receiving payment status.

1. Simple front-end snippet: copy the Javascript (JS) code from merchant admin and paste it on the merchant receipt page (which shows payment success/failed), preferable in the HTML header, before </head> tag.

2. Advanced back-end scripting: merchant is to echo back all the POST variables with one additional variable, i.e. "*treq*" with value 1. PHP sample code is provided below.
   URL: https://pay.merchant.razer.com/RMS/API/chkstat/returnipn.php

Step 3: Merchant to prepare a **notification URL** and **callback URL** script, which is similar to return URL script but serves at the backend, in order to receive consequent payment notification in case the merchant system misses the first notification attempt from the payment gateway.

Example of back-end IPN script for PHP (combined with return URL script)

```php
<?php

$vkey ="xxxxxxxxxx"; //Replace xxxxxxxxxx with Secret_Key

$_POST[treq]   =    1; // Additional parameter for IPN

// Value always 1. Do not change this value.
$tranID    =        $_POST['tranID'];
$orderid   =        $_POST['orderid'];
$status    =        $_POST['status'];
$domain    =        $_POST['domain'];
$amount    =        $_POST['amount'];
$currency          =        $_POST['currency'];
$appcode           =        $_POST['appcode'];
$paydate   =        $_POST['paydate'];
$skey          =    $_POST['skey'];

/*************************************************************
* Snippet code in purple color is the enhancement required
* by merchant to add into their return script in order to
* implement backend acknowledge method for IPN
*************************************************************/
while ( list($k,$v) = each($_POST) ) {
  $postData[]= $k."=".$v;
}
$postdata          = implode("&",$postData);
$url               = "https://pay.merchant.razer.com/RMS/API/chkstat/returnipn.php";
$ch                = curl_init();
curl_setopt($ch, CURLOPT_POST                    , 1          );
curl_setopt($ch, CURLOPT_POSTFIELDS              , $postdata );
curl_setopt($ch, CURLOPT_URL                     , $url       );
curl_setopt($ch, CURLOPT_HEADER                  , 1          );
curl_setopt($ch, CURLINFO_HEADER_OUT             , TRUE       );
curl_setopt($ch, CURLOPT_RETURNTRANSFER     , 1           );
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER     , FALSE      );
curl_setopt($ch, CURLOPT_SSLVERSION         , 6          );  // use only TLSv1.2
$result = curl_exec( $ch );
curl_close( $ch );

/*************************************************************
* To verify the data integrity sending by PG
*************************************************************/
$key0 = md5( $tranID.$orderid.$status.$domain.$amount.$currency );

$key1 = md5( $paydate.$domain.$key0.$appcode.$vkey );

if( $skey != $key1 ) $status= -1; // Invalid transaction
if ( $status == "00" ) {
  if ( check_cart_amt($orderid, $amount) ) {
    // write your script here .....
  }
} else {
  // failure action
}

?>
```

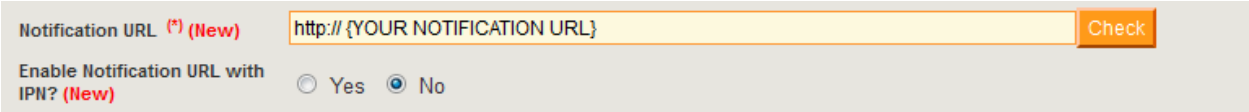
Notification URL with IPN

Direct payment status notification is a back-end instant update mechanism that sends over the payment status notification from gateway directly to the merchant server or system, without relying on any user agent such as web browser or mobile application.

Notification URL script is similar to return URL script but no output is required for front-end user interface. This is also known as background URL.

Implementation:

Step 1: Login to merchant admin and go to merchant profile, fill in the Notification URL. Notification URL with IPN can be activated for better merchant-payment system communication. Choose "Yes" to "Enable Notification URL with IPN", as shown:-



Step 2: Merchant to prepare a notification URL script, which is similar to return URL script.

Step 3: If IPN is disable please ignore this step. Merchants have to echo back all the POST variables with one additional variable, i.e. "*treq*" with value 1. PHP sample code is provided below.
URL: https://pay.merchant.razer.com/RMS/API/chkstat/returnipn.php

Notification Parameters (via POST method)

Variable / Parameter	Type Format / Max Length	Description / Example
nbcb	1 digit numeric	Always equal to **2**, which indicates this is a notification from PG
amount	2 decimal points numeric value	The total amount paid or to be paid for Razer Cash payment request
orderid	alphanumeric, 32 characters	Invoice or order number from merchant system
tranID	integer, 10 digits	Unique transaction ID for tracking purpose
domain	alphanumeric, 32 chars	Merchant ID in PG system
status	2-digit numeric value	00 for Successful payment 11 for failed status
appcode	alphanumeric, 16 chars	Bank approval code. Mandatory for card payment. Certain channels return empty value.

error_code	alphanumeric	Refer to the Error Codes section.
error_desc	text	Error message or description.
skey	32 chars hexadecimal string	This is the data integrity protection hash string. Refer *skey* section for details.
currency	2 or 3 chars (ISO-4217) currency code	Default currency is MYR (indicating Malaysia Ringgit) for Malaysia channels
channel	predefined string in system	Channel references for merchant system
paydate	Date/Time(YYYY-MM-DD HH:mm:ss)	Date/Time of the transaction.
extraP*	optional (on request)	JSON encoded string or array token: 16-digit card token for merchant to store for recurring MIT (merchant initiated transaction) fraudscreen: 1-digit integer, i.e. 1=Unknown, 2=Passed, 3=Alert, 4=Suspicious, 5=Fraud fpx_txn_id: FPX transaction ID fpx_buyer_name: FPX buyer name ccbrand: Visa, MasterCard, AMEX cclast4: Last 4-digit of card number cctype: Credit, Debit, Prepaid ProcessorResponseCode : Authorization response code ProcessorCCVResponse : Credit Card validation response code AvsStreet : Street validation status AvsZip : Zip/Postcode validation status AvsResponse : Association validation status ECI : Electronic commerce indicator

*Note: Values are not URL encoded

Example of Notification URL with IPN script for PHP

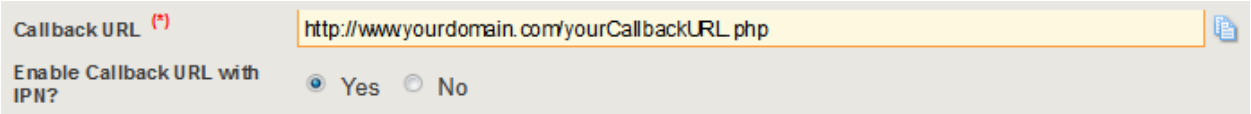
```php
<?php

$vkey ="xxxxxxxxxx"; //Replace xxxxxxxxxx with Secret_Key

$_POST[treq]    =     1; // Additional parameter for IPN. Value always set to 1.

/*********************************
*Don't change below parameters
*********************************/
$nbcb           =     $_POST['nbcb'];
$tranID    =          $_POST['tranID'];
$orderid   =          $_POST['orderid'];
$status    =          $_POST['status'];
$domain    =          $_POST['domain'];
$amount    =          $_POST['amount'];
$currency              =          $_POST['currency'];
$appcode               =          $_POST['appcode'];
$paydate   =          $_POST['paydate'];
$skey           =     $_POST['skey'];

/************************************************************
* Snippet code in purple color is the enhancement required
* by merchant to add into their notification script in order to
* implement backend acknowledge method for IPN
************************************************************/
while ( list($k,$v) = each($_POST) ) {
  $postData[]= $k."=".$v;
}
$postdata              = implode("&",$postData);
$url                   = "https://pay.merchant.razer.com/RMS/API/chkstat/returnipn.php";
$ch                    = curl_init();
curl_setopt($ch, CURLOPT_POST                                  , 1       );
curl_setopt($ch, CURLOPT_POSTFIELDS                            , $postdata );
curl_setopt($ch, CURLOPT_URL                                   , $url     );
curl_setopt($ch, CURLOPT_HEADER                          , 1       );
curl_setopt($ch, CURLINFO_HEADER_OUT          , TRUE     );
curl_setopt($ch, CURLOPT_RETURNTRANSFER     , 1       );
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER       , FALSE    );
curl_setopt($ch, CURLOPT_SSLVERSION           , 6       );  // use only TLSv1.2
$result = curl_exec( $ch );
curl_close( $ch );

/************************************************************
* To verify the data integrity sending by PG
************************************************************/
$key0 = md5( $tranID.$orderid.$status.$domain.$amount.$currency );

$key1 = md5( $paydate.$domain.$key0.$appcode.$vkey );

if( $skey != $key1 ) $status= -1; // Invalid transaction
if ( $status == "00" ) {
  if ( check_cart_amt($orderid, $amount) ) {
    // write your script here .....
  }
} else {
  // failure action
}

?>
```

Callback URL with IPN

Callback mechanism is a back-end activity that is initiated by payment gateway to notify the merchant system once there are changes on any payment status.

Callback URL script is similar to return URL script. Both handle the payment status result; Unlike the return URL script, callback URL script is to handle defer status update or all other non-realtime payment status updates, such as Razer Cash payment.

Merchant must fill the Callback URL in merchant admin in order to get those non-realtime status update from the gateway. Callback URL IPN can be activated for better merchant-payment system communication. Instead of returning all parameters from the gateway, merchant's callback script just need to echo "**CBTOKEN:MPSTATOK**", in plaintext, without double quotes or any HTML tags.



Merchant can actually use the same script for both callback URL & return URL, because there is only one extra parameter in callback request, in addition to return URL parameters, which is "*nbcb*". Besides, an additional status code, 22 has been introduced for "pending" transaction status.

Once there is a status changed event, payment gateway will POST the following parameters to mechant callback URL. If the payment gateway could not get the actual ACK message from the merchant, it will retry for a maximum 3 times, of 15 minutes interval for each attempt.

Callback Parameters (via POST method)

Variable / Parameter	Type Format / Max Length	Description / Example
nbcb	1 digit numeric	Always equal to 1, which indicates this is a callback notification from PG
amount	2 decimal points numeric value	The total amount paid or to be paid for Razer Cash payment request
orderid	alphanumeric, 32 characters	Invoice or order number from merchant system.
tranID	integer, 10 digits	Unique transaction ID for tracking purpose
domain	alphanumeric, 32 chars	Merchant ID in PG system
status	2-digit numeric value	00 for Successful payment 11 for failed status
appcode	alphanumeric, 16 chars	Bank approval code. Mandatory for card payment. Certain channel returns empty value

error_code	alphanumeric	Refer to the Error Codes section
error_desc	text	Error message or description
skey	32 chars hexadecimal string	This is the data integrity protection hash string. Refer *skey* section for details
currency	2 or 3 chars (ISO-4217) currency code	Default currency is MYR (indicating Malaysia Ringgit) for Malaysia channels
channel	predefined string in system	Channel references for merchant system
paydate	Date/Time(YYYY-MM-DD HH:mm:ss)	Date/Time of the transaction
extraP*	optional (on request) - for credit channel	JSON encoded string or array token: 16-digit card token or Direct Debit token for merchant to store for recurring MIT (merchant initiated transaction) fraudscreen: 1-digit integer, i.e. 1=Unknown, 2=Passed, 3=Alert, 4=Suspicious, 5=Fraud fpx_txn_id: FPX transaction ID fpx_buyer_name: FPX buyer name ccbrand: Visa, MasterCard, AMEX cclast4: Last 4-digit of card number cctype: Credit, Debit, Prepaid
	- for Cash-711 channel	store_id: 7eleven store id store_name: 7eleven store name store_address: 7eleven store address

*Note: Values are not URL encoded

Example of callback URL script for PHP

```php
<?php

$vkey ="xxxxxxxxxx"; //Replace xxxxxxxxxx with Secret_Key

$nbcb           =       $_POST['nbcb'];
$tranID     =           $_POST['tranID'];
$orderid    =           $_POST['orderid'];
$status     =           $_POST['status'];
$domain     =           $_POST['domain'];
$amount     =           $_POST['amount'];
$currency               =           $_POST['currency'];
$appcode                =           $_POST['appcode'];
$paydate    =           $_POST['paydate'];
$skey           =       $_POST['skey'];

/*************************************************************
* To verify the data integrity sending by PG
*************************************************************/
$key0 = md5( $tranID.$orderid.$status.$domain.$amount.$currency );
$key1 = md5( $paydate.$domain.$key0.$appcode.$vkey );
if( $skey != $key1 ) $status= -1; // Invalid transaction

if ( $status == "00" ) {
  if ( check_cart_amt($orderid, $amount) ) {
    // write your script here .....
  }
} else {
  // failure action
  // write your script here .....
}

if ( $nbcb==1 ) {
  //callback IPN feedback to notified PG
  echo "CBTOKEN:MPSTATOK"; exit;
}else{
  //normal IPN and redirection
}
?>
```

TL;DR?

SUMMARY ON INTEGRATION, 3 ENDPOINTS AND IPN

INTEGRATION

1) Hosted payment page or normal integration: HTTP POST/GET with all required parameters, either to default payment channel page or to specific channel page, merchant can allow or disallow the buyer to change payment method (PDF)

2) Seamless Integration: popup page or lightbox on merchant checkout flow, as simple as modifying the JavaScript snippet that could be found on GitHub

3) Mobile xdk: support variety of mobile development platforms and available on GitHub

4) Direct Server API: server-to-server request that allows merchant to handle all the UI/UX with higher flexibility and controllability (PDF)

5) Recurring API: server-to-server request that allows merchant initiated transaction (MIT) to debit buyer account anytime (PDF)

6) Offline Payment API: For in-store e-wallet payment acceptance on POS/terminal (PDF)

7) ISO Message Interface: For card acceptance terminal vendor (restricted PDF)

3 ENDPOINTS for payment response notification (For integration 1-6 only)

1) Return URL: realtime web browser or frontend direction endpoint for hosted page, seamless integration, and shopping cart module

2) Notification URL: real-time server-to-server or backend endpoint for all kind of integrations

3) Callback URL: defer update or callback endpoint on non-real time payment such as Razer Cash

IPN

1) **Frontend** IPN, applicable to return URL endpoint only: using JavaScript, you may copy from the snippet code from your merchant portal, note that **treq=0**

2) **Backend** IPN for return URL and notification URL: post back all parameters with additional parameter **treq=1** to **returnipn.php**

3) IPN for **callback** URL: just echo "**CBTOKEN:MPSTATOK**", without the quote(")

Comparison Chart

For **online** payment, merchant to decide which approaches fit the business requirements

Integration approach	**Hosted payment page**	**Seamless integration**	**Inpage checkout**	**Mobile xdk**	**Direct server API + CSE**	**Recurring API**
Browser dependency	Yes, popup and redirection	Yes, popup and redirection	Yes, iframe, popup and redirection	Smartphone built-in browser	Depends	No
UI/UX	Moderate	Better	Better	Best for mobile	Handled by merchant	Handled by merchant
Time to market (man-day)	1-7	2-14	2-14	2-14	5-60	10-90
Suitable for	Fast and easy deployment	Better UX	Better UX	In-app purchase	Better UI & UX	Subscription or billing
Mobile readiness	RWD	RWD	RWD	Native / hybrid	Backend only	Backend only
PCI-DSS compliant	PG	PG	PG	PG	PG / Merchant if handling PAN	PG / Merchant if handling PAN
Availability on github	20/22 of the cart modules	6/22 of the cart modules	Yes	12 dev-tools supported	nil	nil
Available endpoints	all 3 endpoints	all 3 endpoints	all 3 endpoints	notification & callback URL	all 3 endpoints	notification & callback URL
Fraud screening	By PG	By PG	By PG	By PG	By merchant	By merchant

For response handling, setup these endpoints(webhook) to capture the payment response

Response endpoint	**Return URL**	**Notification URL**	**Callback URL**
Browser dependency	Yes	No	No
Payment type response	Realtime payment	Realtime payment	Cash payment / defer update
Reliability	Low	High	High
Security	Low if payment verification is not enabled	High	High
IPN implementation	Frontend: easier to implement using JS snippet with treq=0 Backend: POST back all values + treq=1 to returnipn.php, more reliable	Backend: POST back all values + treq=1 to returnipn.php	Backend: echo “CBTOKEN: MPSTATOK”

MERCHANT REQUEST APIs

PG has prepared plenty of merchant tools for merchants to initiate status requery and cancellation/void of transaction. However, merchants are not allowed to send in requests too frequently (maximum 1 query every 5 seconds). Massive incoming query will lead to IP blocking without prior notice. There will be a validity duration to initiate status requery API call, generally within **1 hour** after the initiated time for online payment and the due time for offline payment.

The back-end services available are:-

1. Direct Status Requery - merchant send status query to processing bank directly
2. Indirect Status Requery - merchant send status query to PG system
3. Daily Transaction Report (Reconciliation) - list all transactions of a specific date
4. Settlement Report (Reconciliation) - settlement details
5. Capture Request - to capture any pre-auth or authorized transaction
6. Reversal Request - merchant to cancel or void a transaction or refund of payment
7. Partial Refund - for merchant who requires partial refund only
8. Partial Refund Status Inquiry by Txn ID
9. Channel Status API
10. Card BIN information API
11. Foreign Exchange Rate API
12. Void Pending-Cash API
13. Recurring Plans API

Notes:

W.e.f 19th October 2017, Razer Merchant Services will migrate all none payment-flow related service APIs to a new FQDN: https://api.molpay.com (and now changed to https://api.merchant.razer.com)
from the original https://www.onlinepayment.com.my

W.e.f 1st November 2017, Razer Merchant Services will shutdown all none payment-flow related service APIs on https://www.onlinepayment.com.my

For sandbox environment, use https://sandbox.merchant.razer.com

Direct Status Requery

This will trigger a query to the payment channel or bank status server and there are cases that bank status server is not in-sync with its payment server that might give different results, that leads to a defer update and will trigger a callback from PG server, once the status is synced and changed.

Request

URL: https://api.merchant.razer.com/RMS/API/gate-query/index.php

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
amount	2 decimal points numeric value	The payment amount
txID	integer, 10 digits	Unique transaction ID for tracking purpose.
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer, obsoleted in new API sets	0 = plain text result (default) 1 = result via POST method

Response

Variable / Parameter	Type Format / Max Length	Description / Example
Amount	2 decimal points numeric value	The payment amount
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
Domain	alphanumeric, 32 chars	Merchant ID in PG system.
Channel	alphanumeric, 100 chars	Payment via Channel
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	alphanumeric	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
Currency	3 chars ISO-4217 currency code	The payment currency. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR
ErrorCode	alphanumeric	Error code defined by channel for failed transactions only
ErrorDesc	alphanumeric	Error description defined by channel for failed transactions only

Formula of skey & VrfKey

skey =md5(txID & domain & verify_key & amount)
VrfKey=md5(Amount & secret_key & Domain & TranID & StatCode)

Example of Direct Status Requery for PHP

```
<?php

$skey = md5($txID . $domain. "xxxxxxxxxx" . $amount);
//Replace xxxxxxxxxx with Verify Key

echo "<a
href='https://api.merchant.razer.com/RMS/API/gate-query/index.php?amount=3899&txID=65234&domain=shopA&skey=e1c4c6
0c99116fffc3ce77bd5fd0f7b1'>
Check payment status for tran ID 65234 </a>";

?>
```

Example of response

type=0 (default output, plain text with linebreaks)	type=1 (POST result sent to URL)
StatCode=00 StatName=captured TranID=65234 Amount=3899.00 Domain=shopA Channel=fpx VrfKey=456cf69e5bddfe8ed47371096 Currency=MYR ErrorCode= ErrorDesc=	$_POST [StatCode] => "00"; $_POST [StatName] => "captured"; $_POST [TranID] => "65234"; $_POST [Amount] => "3899.00"; $_POST [Domain] => "shopA"; $_POST[Channel] => "fpx"; $_POST[VrfKey:]=> "456cf69e5bddfe8ed47371096"; $_POST[Currency] => "MYR"; $_POST[ErrorCode] => ""; $_POST[ErrorDesc] => "";

Indirect Status Requery

There are several types of status requery on PG system:-

1. Query by unique transaction ID (recommended)
2. Query by order ID & get latest matched result (single output) (not recommended)
3. Query by order ID & get all matched results (batch output) (strongly not recommended)
4. Query by multiple order ID (batch output) (strongly not recommended)
5. Query by multiple transaction ID (batch output) (strongly not recommended)

For bulk requery, best practice is to utilize the 4th or 5th APIs once every half an hour for unknown status transactions only

1. Query by unique transaction ID

Request

URL: https://api.merchant.razer.com/RMS/q_by_tid.php

Method: POST or GET

* Request & Response parameters are the same as Direct Status Requery but the format and parameters order of the responses are slightly different.

Example of response

type=0 (default output, plain text with linebreaks)	type=1 (POST result sent to URL)
StatCode: 00 StatName: captured TranID: 10645406 Amount: 138.99 Domain: ShopB VrfKey: 9862acf1099b625c00b225887e715861 Channel: credit OrderID: ABC123 Currency: MYR ErrorCode: ErrorDesc:	$_POST [StatCode] => “00”; $_POST [StatName] => “captured”; $_POST [TranID] => “10565234”; $_POST [Amount] => “3899.00”; $_POST [Domain] => “shopC”; $_POST[VrfKey:] => “456cf69e5bddfe8ed47371096”; $_POST[Channel] => “credit”; $_POST[OrderID] => “ABC123”; $_POST[Currency] => “MYR”; $_POST[ErrorCode] => “”; $_POST[ErrorDesc] => “”;

2. Query by order ID (single output)

Request

URL: https://api.merchant.razer.com/RMS/query/q_by_oid.php

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
amount	2 decimal points numeric value	The payment amount
oID	alphanumeric, 32 chars	Merchant order ID, which might be duplicated.
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer	0 = plain text result (default) 1 = result via POST method
req4token	optional, 1-digit integer	0 = No (default) 1 = Yes for more card related information

Response

Variable / Parameter	Type Format / Max Length	Description / Example
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	alphanumeric	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
OrderID	alphanumeric, 32 chars	Invoice or order number from merchant system.
Amount	2 decimal points numeric value	The payment amount
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
Domain	alphanumeric, 32 chars	Merchant ID in PG system.
BillingDate	date (YYYY-MM-DD HH:mm:ss)	Transaction date/time
BillingName	alphanumeric, 128 chars	Buyer full name
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
Channel	alphanumeric, 100 chars	Payment via channel
Currency	3 chars ISO-4217 currency code	The payment currency. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR
ErrorCode	alphanumeric	Error code defined by channel for failed transactions only

ErrorDesc	alphanumeric	Error description defined by channel for failed transactions only
token	optional, with req4token=1	Card payment only: if PAN has been tokenized
ccbrand	optional, with req4token=1	Card payment only: Visa, MasterCard, AMEX
cclast4	optional, 4-digit numeric with req4token=1	Card payment only: Last 4-digit of PAN
cctype	optional, with req4token=1	Card payment only: Credit, Debit, Prepaid

Formula of skey & VrfKey

skey =md5(oID & domain & verify_key & amount)
VrfKey=md5(Amount & secret_key & Domain & OrderID & StatCode)

3. Query by order ID (batch output)

Request

URL: https://api.merchant.razer.com/RMS/query/q_oid_batch.php

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
oID	alphanumeric, 32 chars	Merchant order ID, which might be duplicated.
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer	0 = plain text result (default) 1 = result via POST method
format	optional, 1-digit integer, for type=1 only	0 = result string with delimiter (|) 1 = result in array
req4token	optional, 1-digit integer	0 = No (default) 1 = Yes for more card related information

Response

Variable / Parameter	Type Format / Max Length	Description / Example
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
BillingDate	date (YYYY-MM-DD HH:mm:ss)	Transaction date
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	alphanumeric	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
Amount	2 decimal points numeric value	The payment amount
BillingName	alphanumeric, 128 chars	Buyer full name
Currency	3 chars ISO-4217 currency code	The payment currency. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR
ErrorCode	alphanumeric	Error code defined by channel for failed transactions only
ErrorDesc	alphanumeric	Error description defined by channel for failed transactions only
token	optional, with req4token=1	Card payment only: if PAN has been tokenized
ccbrand	optional, with req4token=1	Card payment only: Visa, MasterCard, AMEX

cclast4	optional, 4-digit numeric, with req4token=1	Card payment only: Last 4-digit of PAN
cctype	optional, with req4token=1	Card payment only: Credit, Debit, Prepaid

Formula of skey

skey =md5(oID & domain & verify_key)

Example of response

type=0, plain text output, newline with single or two TAB character(s)	TranID BillingDate StatCode StatName Amount BillingName Currency ErrorCode ErrorDesc 418607 2009-11-26 22 pending 25.00 Lenka MYR 418603 2009-11-26 00 captured 125.10 Mika MYR 418583 2009-11-26 00 captured 71.10 Ciara MYR
type=1, format=0, POST variables with delimiter "|"	$_POST[TranID] = "418607|418603|418583"; $_POST[BillingDate] = "2009-11-26|2009-11-26|2009-11-26"; $_POST[StatCode] = "22|00|00"; $_POST[StatName] = "pending|captured|captured"; $_POST[Amount] = "25.00|125.10|71.10"; $_POST[BillingName] = "Lenka|Mika|Ciara"; $_POST[Currency] = "MYR|MYR|MYR"; $_POST[ErrorCode] = "||"; $_POST[ErrorDesc] = "||";
type=1, format=1, POST variables in array	$_POST**[0]**[TranID] = "418607"; $_POST**[0]**[BillingDate] = "2009-11-26"; $_POST**[0]**[StatCode] = "22"; $_POST**[0]**[StatName] = "pending"; $_POST**[0]** [Amount] = "25.00"; $_POST**[0]** [BillingName] = "Lenka"; $_POST**[0]** [Currency] = "MYR"; $_POST**[0]** [ErrorCode] = ""; $_POST**[0]** [ErrorDesc] = ""; $_POST**[1]** [TranID] = "418603"; $_POST**[1]** [BillingDate] = "2009-11-26"; $_POST**[1]** [StatCode] = "00"; $_POST**[1]** [StatName] = "captured"; $_POST**[1]** [Amount] = "125.10"; $_POST**[1]** [BillingName] = "Mika"; $_POST**[1]** [Currency] = "MYR"; $_POST**[1]** [ErrorCode] = ""; $_POST**[1]** [ErrorDesc] = ""; $_POST**[2]** [TranID] = "418583"; $_POST**[2]** [BillingDate] = "2009-11-26"; $_POST**[2]** [StatCode] = "00"; $_POST**[2]** [StatName] = "captured"; $_POST**[2]** [Amount] = "71.10"; $_POST**[2]** [BillingName] = "Ciara"; $_POST**[2]** [Currency] = "MYR"; $_POST**[2]** [ErrorCode] = ""; $_POST**[2]** [ErrorDesc] = "";

4. Query by multiple order ID (batch output)

Request

URL: https://api.merchant.razer.com/RMS/query/q_by_oids.php

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
oIDs	alphanumeric, up to 100 orders	Merchant order ID, must be URLencoded.
delimiter	single character, default is "|"	Avoid using any symbol that might exist in order ID, and also any of these: ",%, *, <, >, ? , \, $, &, =
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer	0 = plain text result (default) 1 = result via POST method
format	optional, 1-digit integer, apply for type=1 only	result string with TAB-newline (default) 0 = result string with delimiter (|) 1 = result in array
req4token	optional, 1-digit integer	0 = No (default) 1 = Yes for more card related information

Response

Variable / Parameter	Type Format / Max Length	Description / Example
OrderID	alphanumeric, 32 chars	Merchant order ID for tracking purpose.
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
BillingDate	date (YYYY-MM-DD HH:mm:ss)	Transaction date
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	alphanumeric	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
Amount	2 decimal points numeric value	The payment amount
BillingName	alphanumeric, 128 chars	Buyer full name
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
Channel	alphanumeric, 100 chars	Payment via channel
Currency	3 chars ISO-4217 currency code	The payment currency. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR

ErrorCode	alphanumeric	Error code defined by channel for failed transactions only
ErrorDesc	alphanumeric	Error description defined by channel for failed transactions only
token	optional, with req4token=1	Card payment only: if PAN has been tokenized
ccbrand	optional, with req4token=1	Card payment only: Visa, MasterCard, AMEX
cclast4	optional, 4-digit numeric, with req4token=1	Card payment only: Last 4-digit of PAN
cctype	optional, with req4token=1	Card payment only: Credit, Debit, Prepaid

Formula of skey and VrfKey

skey =md5(domain & oIDS & verify_key)
VrfKey=md5(Amount & secret_key & Domain & OrderID & StatCode)

Example of response

type=1 format=1	[oid1] => Array ([TranID] => 9994238 [BillingDate] => 2016-11-28 16:32:08 [StatCode] => 00 [StatName] => captured [Amount] => 30.00 [BillingName] => kimyoon [VrfKey] => 311d72c16e0d3b3fc7994ae93467a2d9 [Channel] => mb2u [Currency] => MYR [ErrorCode] => [ErrorDesc] =>) [oid2] => Array ([TranID] => 10004613 [BillingDate] => 2016-11-29 09:56:02 [StatCode] => 11 [StatName] => failed [Amount] => 58.60 [BillingName] => MohdAli [VrfKey] => f9f06b47e23410e624df5e272accb27dc [Channel] => fpx [Currency] => MYR [ErrorCode] => [ErrorDesc] =>) ... [oidN] => Array ([TranID] => - [BillingDate] => - [StatCode] => - [StatName] => - [Amount] => - [BillingName] => - [VrfKey] => - [Channel] => - [Currency] => - [ErrorCode] => - [ErrorDesc] => -)
type=1 format=0 delimiter=|	[OrderID] => oid1|oid2|...|oidN [TranID] => 9994238|10004613|...|- [BillingDate] => 2016-11-28 16:32:08|2016-11-29 09:56:02|...|- [StatCode] => 00|11|...|- [StatName] => captured|failed|...|- [Amount] => 30.00|58.60|...|- [BillingName] => Nurbaizura|KUMARASAN|...|- [VrfKey] => 311d72c16e0d3b3fc7994ae93467a2d9|f9f06b47e23410e624df5e272accb27dc|...|- [Channel] => mb2u|fpx|...|- [Currency] => MYR|MYR|...|- [ErrorCode] => ||...|- [ErrorDesc] => ||...|-

5. Query by multiple transaction ID (batch output)

Request

URL: https://api.merchant.razer.com/RMS/query/q_by_tids.php

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
tIDs	concatenated transaction ID with "|" up to 100 items	A group of transaction ID, must be URLencoded
domain	alphanumeric, 32 chars	Merchant ID in PG system
skey	32 chars hexadecimal string	This is the data integrity protection hash string
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer	0 = plain text result (default) 1 = result via POST method
format	optional, 1-digit integer, apply for type=1 only	0 = result string with delimiter (|) 1 = result in array
req4token	optional, 1-digit integer	0 = No (default) 1 = Yes for more card related information

Response

Variable / Parameter	Type Format / Max Length	Description / Example
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
Amount	2 decimal points numeric value	The payment amount
BillingDate	date (YYYY-MM-DD HH:mm:ss)	Transaction date
BillingName	alphanumeric, 128 chars	Buyer full name
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	alphanumeric	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
OrderID	alphanumeric, 32 chars	Merchant order ID for tracking purpose.
Currency	3 chars ISO-4217 currency code	The payment currency. E.g. MYR, USD, EUR, AUD, SGD, CNY, IDR
ErrorCode	alphanumeric	Error code defined by channel for failed transactions only

ErrorDesc	alphanumeric	Error description defined by channel for failed transactions only
token	optional, with req4token=1	Card payment only: if PAN has been tokenized
ccbrand	optional, with req4token=1	Card payment only: Visa, MasterCard, AMEX
cclast4	optional, 4-digit numeric, with req4token=1	Card payment only: Last 4-digit of PAN
cctype	optional, with req4token=1	Card payment only: Credit, Debit, Prepaid

Formula of skey and VrfKey

skey =md5(domain & tIDS & verify_key)
VrfKey=md5(Amount & secret_key & Domain & TranID & StatCode)

Daily Transaction Report (Reconciliation)

PG Daily Transaction Report provides merchant end-of-day (EoD) reconciliation or to verify all transactions for a specific date.

Request

URL: https://api.merchant.razer.com/RMS/API/PSQ/psq-daily.php

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
merchantID	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
rdate	date (YYYY-MM-DD) or date(YYYY-MM-DD HH:ii:ss)	The date or beginning time of transactions to query 2020-10-10 or 2020-10-10 07:11:24
rduration	optional, second in numeric	Without rduration, the search duration is 24 hours or 86400 seconds starting from rdate
status	optional, alphanumeric, 32 chars	00 - success 11 - failed 22 - pending Combine with delimiter "|" for multiple status or left empty for all status.
version	version	2 or 3
additional_fields [New from Version 2]	optional, predefined tags	BillingEmail - billing email TransactionRate - transaction rate BillingInfo - billing info TransactionCost - transaction cost Channel - channel BillingMobileNumber - billing mobile TransactionFee - transaction fee GST - GST (tax) NetAmount - net amount IPAddress - IP address BankName - bank name BIN - card no (hidden) ExpiryDate - card expiry date StatusDescription - status description SettlementDate - settlement date PaidDate - paid date TerminalID - terminal ID PayTransactionID - transaction ID BuyerName - buyer name CaptureRefID - Capture Reference ID (Only in v3) RefundRefID - Refund Reference ID (Only in v3) **all** - for all additional fields Combine with delimiter "," for multiple fields
response_type [New from Version 2]	optional	Response format in either text/json/csv(only in v2 and v3)

Formula of skey

skey =md5(rdate & merchantID & secret_key)

Response

(plain text with newline & TAB characters)

Variable / Parameter	Type Format / Max Length	Description / Example
BillingDate	date/time (YYYY-MM-DD HH:mm:ss)	Transaction date/time
OrderID	alphanumeric, 32 chars	Invoice or order number from merchant system.
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
Channel	Predefined string in PG system	Channel references for the merchant system.
Amount	2 decimal points numeric value	The payment amount
StatCode	2-digit numeric	00 = Success 11 = Failure 22 = Pending
StatName	word	Success: captured, settled, authorized Failure: failed, cancelled, chargeback, release, reject/hold, blocked, ReqCancel, ReqChargeback Pending: Pending, Unknown
BillingName	alphanumeric, 128 chars	Buyer full name
ServiceItem	text	Billing Information / Description (newline will be replaced by whitespace)
Additional fields: BillingEmail TransactionRate TransactionCost BillingMobileNumber TransactionFee GST NetAmount IPAddress BankName ExpiryDate StatusDescription SettlementDate PaidDate TerminalID PayTransactionID BuyerName	text (default) or JSON string	BillingEmail = ABC@razer.com TransactionRate = 0.0290 TransactionCost = 0.0000 BillingMobileNumber = 0123456789 TransactionFee = 100.0 GST = 0 NetAmount = 1900.0 IPAddress = 192.168.0.1 BankName = unknownbank ExpiryDate = 2020 StatusDescription = This is status description SettlementDate = 2020-05-26 10:51:51 PaidDate = 2020-05-31 10:51:51 TerminalID = 999

Settlement Report (Reconciliation)

PG Settlement Report provides merchants the fund transfer or balance clearance and settlement reconciliation for a specific date.
Version 1.0, 2.0 and 3.0 will be obsoleted and the following is the latest specification of version 4.0. Version 4.0 will include refund and chargeback records that are excluded from the settlement batch.

Request

URL: https://api.merchant.razer.com/RMS/API/settlement/report.php

Method: GET

Variable / Parameter	Type Format / Max Length	Description / Example
version	Integer, Default value is 1.0.	Indicate version of the API. Current version is 5.0. If not specified, version 1.0 will be used.
merchant_id	alphanumeric, 32 chars	Merchant ID in PG system
token	32 chars hexadecimal string	This is the data integrity protection hash string
date	date (YYYY-MM-DD)	The date of settlement to query
format	json, xml, csv	Recommend to use csv for large file
download	optional, set to “y” for download mode	Download option is only available for CSV format
page	Integer, Default value is 0 mean display all.	Determines the page to view for D records that have a limit of 5000 transactions. Currently available to version 5.0.

Formula of token

token =md5(merchantID & secret_key & date)

Response

Value	Type Format / Max Length	Description / Example
RecordIdentifier	1 character	H: Header
SettlementCurrency	3 chars ISO-4217 currency code	Default is MYR, Settlement currency
SettlementNetAmount	numeric without decimal and comma	total settlement amount, 5331674 is equal to 53,316.74 in that currency
SettlementCommissionAmount	numeric without decimal and comma	total settlement fees, sum of the transfer fees and 3rd party costs
NumberOfTransactions	numeric	total transaction number of settled transaction or D data type in the report, not including R & G type
BatchReferenceNumber	alphanumeric	Reference number of this settlement

SettlementDate	YYYYMMDD	The settlement date
SettlementGSTAmount	numeric without decimal and comma	Sum of settled transactions GST amount
BankAccount	alphanumeric	Bank swift code and bank account number
RefundNetAmount	numeric without decimal and comma	Sum of the refund fees
RefundGSTAmount	numeric without decimal and comma	Taxes : GST/VAT on the RefundNetAmout

Variable / Parameter	Type Format / Max Length	Description / Example
RecordIdentifier	1 character	D: Data or the content R: Refund / Chargeback Data G: Data (Original txn data - Txn fully refunded before settlement)
MerchantId	alphanumeric	Merchant ID in PG system
OrderId	alphanumeric	Order references for merchant system
Channel	alphanumeric	Payment channel / method
AcquirerReference	alphanumeric	PG transaction ID
RefundID	n{1..11}	Refund ID provided by PG
MerchantRefID	ans{1..100}	Unique refund tracking/reference ID by merchant
RefundFees	numeric without decimal and comma	PG refund fees. D: Always Zero ‘0’ value R: PG refund fees (positive value)
TransactionNetAmount	numeric without decimal and comma	D: amount after deducting MDR/fees. Formula: TransactionGrossAmount - TransactionCommissionAmount R: amount refunded to the buyer (full/partial) after adding Refund fees. Formula: TransactionGrossAmount + RefundFees
TransactionCommissionAmount	numeric without decimal and comma	The MDR or transaction fee or commission D: MDR R: Recalculate MDR for partial refund during captured and charging model have %. Otherwise the value is 0.
TransactionDate	YYYYMMDD	The date of transaction been created
TransactionTime	HHmmss	The time of transaction been requested/created
TransactionGrossAmount	numeric without decimal and comma	D: Payment amount R: amount refunded to the buyer (full/partial)
TransactionCurrency	3 chars ISO-4217 currency code	The currency sent in by merchant or most of the time, paid by the buyer
TransactionGST	numeric without decimal and comma	Taxes : GST/VAT

SettlementNetAmountInProcessingCurrency	numeric without decimal and comma	D: TransactionNetAmount after deducting GST. Formula: TransactionGrossAmount - TransactionCommissionAmount - TransactionGST R: TransactionNetAmount after deducting GST. Formula: NEGATIVE (TransactionGrossAmount + RefundFees + TransactionGST - TransactionCommissionAmount)
SettlementNetAmount	numeric without decimal and comma	Net amount in settlement currency, after forex conversion (round to 2 decimal place). Formula: SettlementNetAmountInProcessingCurrency*Forex
SettlementCurrency	3 chars ISO-4217 currency code	MYR, SGD, USD, ...
Forex	numeric without 8 digit decimal and comma	Forex rate for multi currency transactions. Eg: 1.00000000 = 100000000
Status	alphanumeric	The transaction status. Usually SETTLED for D, REFUND for R, CANCELLED for G

Variable / Parameter	Type Format / Max Length	Description / Example
RecordIdentifier	1 character	F: Record summary
NumberOfTransactionsTotal	integer	Total number of D records
NumberOfTransactionsShown	integer	Total number of D records been display
FilterTransactionsTotalPages	integer	Total available pages for D records
FilterTransactionsLimitPerPage	integer	Limit record been display for D records

***Note: F row only available in *version* 5.0 and when *page* value is more than 0. To speed up the response on the subsequence pages, row R & G will be excluded and may only appear at page 1.**

Error (always in JSON format)

Variable / Parameter	Description / Example
success	false
token	invalid token
date	invalid date format, eg. yyyy-mm-dd

Example (in JSON format)

The following example might not be accurate and please only take it as a format reference


```
[
        {
        "RecordIdentifier": "H",
        "SettlementCurrency": "MYR",
        "SettlementNetAmount": "1033293",
        "SettlementCommissionAmount": "2439",
        "NumberOfTransactions": 15,
        "BatchReferenceNumber": "20170817-613",
        "SettlementDate": "20170817",
        "SettlementGSTAmount": "2239",
        "BankAccount": "(MALAYAN BANKING BHD) MerchantID SDN BHD 5144 8457 3110",
        "RefundNetAmount": "179741",
        "RefundGSTAmount": "334"
        },
        {
    "RecordIdentifier": "D",
    "MerchantId": "merchantIDSB",
    "OrderId": "OP-02559",
    "Channel": "maybank2u",
    "AcquirerReference": "18300981",
    "RefundFees": "0",
    "TransactionNetAmount": "9732",
    "TransactionCommissionAmount": "178",
    "TransactionDate": "20170807",
    "TransactionTime": "112250",
    "TransactionGrossAmount": "9900",
    "TransactionCurrency": "MYR",
    "TransactionGST": "010",
    "SettlementNetAmountInProcessingCurrency": "9722",
    "SettlementNetAmount": "000",
    "SettlementCurrency": "MYR",
    "Forex": null,
    "Status": "SETTLED"
    },
    {
    "RecordIdentifier": "D",
    "MerchantId": "merchantIDSB",
    "OrderId": "163138",
    "Channel": "credit",
    "AcquirerReference": "18310736",
    "RefundFees": "0",
    "TransactionNetAmount": "294521",
    "TransactionCommissionAmount": "9656",
    "TransactionDate": "20170807",
    "TransactionTime": "133001",
    "TransactionGrossAmount": "303630",
    "TransactionCurrency": "MYR",
    "TransactionGST": "547",
    "SettlementNetAmountInProcessingCurrency": "293974",
    "SettlementNetAmount": "000",
    "SettlementCurrency": "MYR",
    "Forex": null,
    "Status": "SETTLED"
    },
```


... segment removed ...

```
        {
    "RecordIdentifier": "D",
    "MerchantId": "merchantIDSB",
    "OrderId": "79126387162378123",
    "Channel": "credit",
    "AcquirerReference": "18516590",
    "RefundFees": "0",
    "TransactionNetAmount": "153745",
    "TransactionCommissionAmount": "5040",
    "TransactionDate": "20170810",
    "TransactionTime": "120057",
    "TransactionGrossAmount": "158500",
    "TransactionCurrency": "MYR",
    "TransactionGST": "285",
    "SettlementNetAmountInProcessingCurrency": "153460",
    "SettlementNetAmount": "000",
    "SettlementCurrency": "MYR",
    "Forex": null,
    "Status": "SETTLED"
    },
    {
    "RecordIdentifier": "D",
    "MerchantId": "merchantIDSB",
    "OrderId": "31283671293",
    "Channel": "credit",
    "AcquirerReference": "18532262",
    "RefundFees": "0",
    "TransactionNetAmount": "9700",
    "TransactionCommissionAmount": "318",
    "TransactionDate": "20170810",
    "TransactionTime": "155958",
    "TransactionGrossAmount": "10000",
    "TransactionCurrency": "MYR",
    "TransactionGST": "018",
    "SettlementNetAmountInProcessingCurrency": "9682",
    "SettlementNetAmount": "000",
    "SettlementCurrency": "MYR",
    "Forex": null,
    "Status": "SETTLED"
    },

        {
    "RecordIdentifier": "R",
    "MerchantId": "merchantIDSB",
    "OrderId": "54671293761293",
    "Channel": "credit",
    "AcquirerReference": "13251784",
    "RefundFees": 0,
    "TransactionNetAmount": "-179741",
    "TransactionCommissionAmount": "000",
    "TransactionDate": "20170503",
    "TransactionTime": "155958",
    "TransactionGrossAmount": "-185300",
    "TransactionCurrency": "MYR",
    "TransactionGST": "-018",
    "SettlementNetAmountInProcessingCurrency": "-179741",
    "SettlementNetAmount": "000",
    "SettlementCurrency": "MYR",
    "Forex": null,
    "Status": "REFUND"
    }
]
```

Unsettled/Refunded Transaction Report (Exclusion from settlement)

Any unsettled transaction that has been cancelled within a given settlement batch will be retrievable using this API

Request

URL: https://api.merchant.razer.com/RMS/API/settlement/report_refund.php

Method: GET

Variable / Parameter	Type Format / Max Length	Description / Example
merchant_id	alphanumeric, 32 chars	Merchant ID in PG system
token	32 chars hexadecimal string	This is the data integrity protection hash string
date	date (YYYY-MM-DD)	The date of settlement to query
format	json, xml, csv	
download	optional, set to “y” if	Download option is only available for CSV format

Formula of token

token =md5(merchantID & secret_key & date)

Response

Value	Type Format / Max Length	Description / Example
H	1 character	Header
Settlement currency	3 chars ISO-4217 currency code	Default is MYR
total settlement amount	numeric without decimal and comma	5331674 is equal to 53,316.74 in that currency
total settlement fees	numeric without decimal and comma	Sum of the transfer fees and 3rd party costs
total refund transaction number	numeric	Total number of refund transactions
batch reference number	alphanumeric	Reference number of this settlement
date	YYYYMMDD	The settlement date
bank account	alphanumeric	Bank swift code and bank account number
Start date	YYYY-MM-DD	The settlement period start date
End date	YYYY-MM-DD	The settlement period end date

Variable / Parameter	Type Format / Max Length	Description / Example
R	1 character	Refund Data
merchant ID	alphanumeric	Merchant ID in PG system
Order ID	alphanumeric	Order references for merchant system
Channel / transaction ID	alpha-numeric / numeric	Payment channel / PG transaction ID
Transaction net amount	Numeric without decimal and comma	This should equal the original transaction amount minus the fee or commission
Transaction commission amount	Numeric without decimal and comma	The MDR or transaction fee or commission
Forex Rate	0	0
Settlement amount	0	0
Settlement currency	0	0
Transaction date	YYMMDD	The date of settlement
Transaction gross amount	Numeric without decimal and comma	Original transaction amount
Transaction currency	3 chars ISO-4217 currency code	Default is MYR
Status	alpha-numeric	The transaction status. Usually REFUND

Error (always in JSON format)

Variable / Parameter	Description / Example
success	false
token	invalid token
date	invalid date format, eg. yyyy-mm-dd

Capture Request (For pre-auth or authorized transaction)

Merchant who uses preauth or authorize payment mode may capture the transaction at a later stage by using this API

Request

URL: https://api.merchant.razer.com/RMS/API/capstxn/index.php

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
domain	alphanumeric, 32 chars	Merchant ID in PG system
tranID	integer, 20 digits	Unique PG transaction ID
amount	2 decimal points numeric value	The total amount paid or to be paid
RefID	Alphanumeric, 100 chars	Reference ID for tracking purpose sent by merchant
skey	32 chars hexadecimal string	This is the data integrity protection hash string

Response (JSON Format)

Variable / Parameter	Type Format / Max Length	Description / Example
TranID	integer, 20 digits	Unique transaction ID for tracking purposes
Domain	alphanumeric, 32 chars	Merchant ID in PG system.
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string
StatCode	2-digit numeric	00 = Success 11 = Failure 12 = Invalid or unmatched security hash string 13 = Not a credit card transaction 15 = Requested day is on settlement day 16 = Forbidden transaction 17 = Transaction not found 18 = Missing required parameter 19 = Domain not found 20 = Temporary out of service 21 = Authorization expired 22 = Not allowed to perform partial capture 99 = General Error(Please check with RMS Support)
StatDate	date (YYYY-MM-DD)	Response date & time
PartialCaptureTranID	integer, 20 digits	Created upon successful partial capture request

Formula of skey & VrfKey

skey =md5(txnID & amount & domain & verify_key)
VrfKey=md5(secret_key & Domain & TranID & StatCode)

Reversal Request

Merchant can request a reversal of transaction via original payment method for an “authorized” **card**, and “captured” **card (including recurring payment), M2U, CIMB Clicks, Hong Leong Connect, Razer Pay, Alipay-Spot, POS Terminal, Boost and WeChat Pay** payment.

For most of the payment channels above, the transaction can be “**void**” immediately on the same day before settlement (card payment at around 10pm and other channels are 11:59pm local time). For a payment channel that accepts refund requests, the transaction that happens within 180 days will be refunded within 7-14 days after the request is sent.

Please note that this API is to send a refund request, but not getting the status of the refund process. All successful requests shall be executed and if there is any exceptional case, PG support team will contact the merchant to resolve the issue.

Request

URL: https://api.merchant.razer.com/RMS/API/refundAPI/refund.php

Method: POST or GET

Variable / Parameter	Type Format / Max Length	Description / Example
txnID	integer, 10 digits	Unique transaction ID for tracking purpose.
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	This is the data integrity protection hash string.
url	optional, URL for POST response	The URL to receive POST response from PG
type	optional, 1-digit integer	0 = plain text result (default) 1 = result via POST method

Response

Variable / Parameter	Type Format / Max Length	Description / Example
TranID	integer, 10 digits	Unique transaction ID for tracking purpose.
Domain	alphanumeric, 32 chars	Merchant ID in PG system.
VrfKey	32 chars hexadecimal string	This is the data integrity protection hash string.
StatCode	2-digit numeric	00 = Success (will proceed the request) 11 = Failure 12 = Invalid or unmatched security hash string 13 = Not a refundable transaction 14 = Transaction date more than 180 days 15 = Requested day is on settlement day 16 = Forbidden transaction 17 = Transaction not found 18 = Duplicate partial refund request

		19 = Merchant not found 20 = Missing required parameter
StatDate	date (YYYY-MM-DD HH:mm:ss)	Response date & time

Formula of skey & VrfKey

skey =md5(txnID & domain & secret_key)
VrfKey=md5(secret_key & Domain & TranID & StatCode)

Reversal Reference Table

Payment Channel	Support Void / Refund / Both	Cut-off time for VOID request	Processing method	Credit to buyer within
MYR card payment	Both	10pm GMT+8	manual	14-business day
Maybank2u	Refund	-	manual	7-business day
CIMB Clicks	Refund	-	manual	7-business day
Hong Leong Connect	Refund	-	manual	7-business day
Razer Pay	Both	11:59pm GMT+8	auto	1-business day
Alipay Spot	Both	11:59pm GMT+8	auto	1-business day
Wechat Pay (CN, MY)	Both	11:59pm GMT+8	auto	1-business day
Boost	Both	11:59pm GMT+8	auto	1-business day
TnG e-Wallet	Both	11:59pm GMT+8	auto	1-business day
Grab Pay	Both	11:59pm GMT+8	auto	1-business day
Maybank QR Push	Refund	-	manual	7-business day

Advanced Full/Partial Refund

Merchants can request a full/partial refund for a “captured” or “settled” transaction regardless of the payment method. The request can be sent within 180 days from the transaction creation date and the refund process will take about 7-14 days after the request is sent.

Merchants can enable the “**Refund Portal**” so that buyers who need a refund could easily provide the bank account details to shorten the refund lead time.

Request

URL: https://api.merchant.razer.com/RMS/API/refundAPI/index.php

Method: POST or GET

Field Name	Data Type(Size)	M/O	Description
RefundType	a{1}	M	P - Partial Refund
MerchantID	an{1..32}	M	Merchant ID provided by PG
RefID	ans{1..100}	M	Unique reference ID for tracking purpose sent by merchant
TxnID	n{1..10}	M	PG Transaction ID
Amount	n{10,2}	M	eg. ‘5.00’ Amount to be refunded
BankCode	an{8}	C	Applicable for Online Banking and Physical Payment transactions only. (Refer to predefined bank lists)
BankCountry	a{2}	C	Applicable for Online Banking and Physical Payment transactions only. Two letters country ISO code. (Default value: MY)
BeneficiaryName	as{1..100}	C	Applicable for Online Banking and Physical Payment transactions only.
BeneficiaryAccNo	ans{1..100}	C	Applicable for Online Banking and Physical Payment transactions only.
Signature	an{32}	M	This is the data integrity protection hash string.
mdr_flag	n{1}	O	This is to include or exclude MDR refund to the buyer if the amount is same as bill amount. Available value is as below: 0 - Include MDR/Full Refund (Default) 1 - Exclude/Reserved MDR
notify_url	as	O	This is the URL for merchant to receive refund status (same response as this API), either 00 (Success) or 11 (Rejected) with slightly different Signature

Signature = md5(RefundType . MerchantID . RefID . TxnID . Amount . **secret_key**)

Response

PG responds JSON format to merchant upon a successful request (positive result)

Field Name	Data Type(Size)	M/O	Description
RefundType	a{1}	M	Echo of merchant request
MerchantID	an{1..32}	M	Echo of merchant request
RefID	ans{1..100}	M	Echo of merchant request
RefundID	n{1..11}	M	Refund ID provided by PG
TxnID	n{1..10}	M	Echo of merchant request
Amount	n{10,2}	M	Echo of merchant request
Status	n{2} [00, 11, 22]	M	22 for 'Pending' (Upon Request) 11 for 'Rejected' (via notify_url) 00 for 'Success' (Refunded, via notify_url)
Signature	a{32}	M	This is data integrity protection hash string.
reason	ans{1..255}	O	Reason for rejected status

Signature = md5(RefundType . MerchantID . RefID . RefundID . TxnID . Amount . Status . **secret_key**)

PG will respond the following in JSON format once error occurs (negative result)

Field Name	Data Type(Size)	Description
error_code	an{5}	Refer to Appendix C
error_desc	ans{1..255}	Refer to Appendix C

Partial Refund Status Inquiry by TxnID/RefID

Merchant is able to do a status inquiry for a refund transaction.

Request Type 1

URL: https://api.merchant.razer.com/RMS/API/refundAPI/q_by_txn.php

Method: POST or GET

Field Name	Data Type(Size)	M/O	Description
TxnID	n{1..10}	M	PG Transaction ID
MerchantID	a{1..32}	M	Merchant ID provided by PG
Signature	n{1..32}	M	This is the data integrity protection hash string.

Signature = md5(TxnID . MerchantID . verify_key)

Request Type 2

URL: https://api.merchant.razer.com/RMS/API/refundAPI/q_by_refID.php

Method: POST or GET

Field Name	Data Type(Size)	M/O	Description
RefID	ans{1..100}	M	Unique tracking/references ID from merchant
MerchantID	a{1..32}	M	Merchant ID provided by PG
Signature	n{1..32}	M	This is the data integrity protection hash string.

Signature = md5(RefID . MerchantID . verify_key)

Response

PG responds JSON format to merchant upon a successful request (positive result)

Field Name	Data Type(Size)	M/O	Description
TxnID	n{1..10}	M	Echo of merchant request
RefID	ans{1..100}	M	Unique tracking/references ID from merchant
RefundID	n{1..11}	M	Refund ID provided by PG
Status	a{8}	M	'pending' , rejected', 'success'
LastUpdate	ans{1..255}	M	Last update timestamp.

Appendix for Partial Refund and Partial Refund Status Inquiry

Appendix A : Data Type Details

Code	Description
a	Letters, A-Za-z
n	Numbers, 0-9
s	Symbols, .:|?*,!&_-
{x}	Fixed length x
{y..x}	Length range: y – x
{y,x}	Number range: 0-9. 0-9

Appendix B : M/O Details

Code	Description
M	Mandatory field.
O	Optional field.
C	Conditional field.

Appendix C : Error Code & Description in JSON

error_code	error_desc
PR001	Refund Type not found.
PR002	MerchantID field is mandatory.
PR003	RefID field is mandatory.
PR004	TxnID field is mandatory
PR005	Amount field is mandatory.
PR006	Signature field is mandatory
PR007	Merchant ID not found.
PR008	Invalid Signature.
PR009	Txn ID not found.
PR010	Transaction must be in authorized/captured/settled status
PR011	Exceed refund amount for this transaction.
PR012	Bank information is not applicable for credit channel transaction.
PR013	BankCode not found in our database, please contact support.

PR014	Bank information is mandatory for non-credit channel transaction.
PR015	Server is busy, try again later.
PR016	Duplicate RefID found, please provide a unique RefID.
PR017	Refund request for transaction that is out of the allowed period.
PR018	BeneficiaryName cannot contain non-alphanumeric characters.
PR019	Refund is not allowed / Only partial refund is allowed / Only full refund is allowed.
PR020	Insufficient balance to refund.
INQ001	TxnID field is mandatory.
INQ002	MerchantID field is mandatory.
INQ003	Signature field is mandatory.
INQ004	Merchant ID not found.
INQ005	Invalid Signature.
INQ006	Unable to find refund transaction.
INQ011	RefID field is mandatory.

Static QR-Code Generator

For merchant to generate static QR code of e-wallet

Request

URL: https://api.merchant.razer.com/RMS/API/staticqr/index.php

Method: POST or GET

Field Name	Data Type(Size)	M/O	Description
merchantID	Alphanumeric, 32 chars	M	Merchant ID provided by PG
channel	Alphanumeric, 32 chars	M	Channel requested: • AlipaySQR - Alipay Static QR • WeChatPaySQR - WeChat Pay Static QR
orderid	Alphanumeric, 32 chars	M	Items ID. E.g: S001
currency	Alphabet, 3 chars	M	ISO-4217 currency code.
amount	Numeric	M	The total amount to be paid in one purchase order. 2 decimal points and comma(,) is not allowed.
bill_name	Alphanumeric, 128 chars	M	Items name.
bill_desc	Text	M	Item description.
checksum	Alphanumeric, 32 chars	M	This is a request integrity protection hash string.

Checksum = md5(merchantID . channel . orderid . currency . amount . verify_key)

Response

PG responds JSON format to merchant upon a successful request

Field Name	Data Type(Size)	M/O	Description
status	Boolean	M	true/false
qrcode_data	URL	M	QR-Code data. Available for status true.
qrcode_link	URL	M	Link to view QR-Code image. Available for status true.
error_code	Alphanumeric	M	Error code when status is false.
error_mesg	Text	M	Error description when status is false.

error_code	error_mesg
SQ001	Missing required parameter (<FieldName>).
SQ002	System is busy now, temporarily out of service. Please try again later.
SQ003	Merchant info not found.

SQ005	Invalid checksum value.
SQ004	Currency not supported.
SQ006	Your account doesn't subscribe to this channel. Please contact our support teams to enable this.

Channel Status API

In order to know the healthiness of payment channels, this API allows system-wide or merchant-only successful rate (OK rate) of a channel in real time for frequently used payment channels. Merchants can always check the latest 1 hour system-wide OK rate of the channel (card payment and internet banking) and a few window sizes, i.e. the latest 1, 3, 6, and 12 hours for merchant-only OK rate for all channels, including Razer Cash. Keep in mind that Razer Cash is not a real-time payment channel and sampling is based on payment request time and not the time that payment is made.

This API gives a worst case scenario result, please DO NOT approach the RMS support team if your number of transactions of a specific payment channel is less than 30 or if there are multiple failure attempts from the same buyer. Try to optimize your OK rate (success rate or SR) if yours are far below the system-wide level. Frequent query on merchant-only OK rate might slow down your payment processing as well.

Prompt alert or warning instead of turning off the channel at 0% or low system-wide OK rate is always a best practice.

Request

URL: https://api.merchant.razer.com/RMS/API/chkstat/OK-rate.php

Method: GET

Variable / Parameter	Type Format / Max Length	Description / Example
domain	alphanumeric, 32 chars	Merchant ID in PG system.
reqTime	YYYYMMDDHHmmss	Request date & time, e.g. 20161202153423
reqType	Global, Merchant	Global: system-wide OK rate Merchant: merchant only OK rate
skey	32 chars hexadecimal string	For merchant access verification purpose
duration	1, 3, 6, 12 (for reqType=merchant only)	Latest sampling window size in **HOUR** for merchant-only request. If the total sampling number is less than 30, it has less reference value due to the low statistical significance and n/a will be given

Response (JSON format)

Variable / Parameter	Type Format / Max Length	Description / Example
StatTime	YYYYMMDDHHmmss	Status date & time, e.g. 20161202153435
OK-rate	{ "Channel 1" : OK_rate1, "Channel 2" : OK_rate2, ...	Channel name with successful rate measure at the *StatTime*. Please refer to "*Direct Server API*" or "*Seamless Integration*" for channel name.

	}	{
		“credit”: 75,
		“maybank2u”: 70,
		“cimbclicks”: 65,
		“fpx”: 56,
		“hlb”: 60,
		“rhb”: 52,
		…
		“fpx_pbb”: 45,
		“fpx_bimb”: 80,
		}
		
		where the OK rate is an integer range from 0-100. Channel might be added/removed without prior notice.

Formula of skey

skey = md5(domain & secret_key & reqTime & reqType)

Card BIN information API

To retrieve the card BIN information such as card type and the issuer information.

Request

URL: https://api.merchant.razer.com/RMS/query/q_BINinfo.php

Method: GET

Variable / Parameter	Type Format / Max Length	Description / Example
domain	alphanumeric, 32 chars	Merchant ID in PG system.
skey	32 chars hexadecimal string	For merchant access verification purpose
BIN	6-digit numeric	First 6-digit number of the PAN, e.g. 519603

Response (JSON format)

Variable / Parameter	Type Format / Max Length	Description / Example
card_brand	VISA / MASTERCARD / AMEX / UP / JCB / DC	MASTERCARD
card_issuer	ans{100}	CIMB BANK BERHAD
debit_credit	DEBIT / CREDIT / PREPAID / N/A	CREDIT
card_type	ans{100}	STANDARD other example like GOLD, PLATINUM, CLASSIC, co-brand name is available
card_country	ISO-3166-1 alpha-2	MY
islamic_card	Y / N	Y

PG will respond the following in JSON format once error occurs (negative result)

Field Name	Data Type(Size)	Description
error_code	an{7}	Refer to Appendix A
error_desc	ans{1..255}	Refer to Appendix A

Formula of skey

```
skey  = md5( domain & secret_key & BIN )
```

Appendix A : Error Code & Description in JSON

error_code	error_desc
QBIN001	Missing required Parameter.
QBIN002	Merchant info not found.
QBIN003	Invalid skey.
QBIN004	Card BIN info not found.
QBIN005	System is busy now, temporarily out of service. Please try again later.

Foreign Exchange Rate API

In order to know the current foreign exchange (FX) of each supported currency, this API allows merchants to know the current exchange rate for each supported currency.

Request

URL: https://api.merchant.razer.com/RMS/query/q_fx_rate.php

Method: GET

Variable / Parameter	Type Format / Max Length	Description / Example
domain	alphanumeric, 32 chars	Merchant ID in PG system.
reqtime	YYYYMMDD	Request date & time, e.g. 20161202
source	optional, 1-digit integer	Predefined value as below: 1. BNM 2. Maybank
skey	32 chars hexadecimal string	For merchant access verification purpose

Response (JSON format)

Variable / Parameter	Type Format / Max Length	Description / Example
base	3 chars ISO-4217 currency code	MYR
reqtime	YYYYMMDD	20170128
rate	float(2,4) in array	1.2888
[currency] under “rate”	3 chars ISO-4217 currency code	Currency to compared with MYR (USD,SGD,etc)

Formula of skey

skey = md5(domain & secret_key & reqTime)

Will list out the list of currency, exchange rate compared to 1 MYR with rate expiration date/time.

```
{
  “base” :  “MYR”
  “reqtime” : “20170128”
  "rate":{
 "USD":{"units":1,"rates":0.24425989252565},
  ...
  }
}
```

Void Pending-Cash API

For merchants to cancel and void the cash payment request order, before getting paid or the expiry time, and force-to-expired.

Request

URL: https://api.merchant.razer.com/RMS/API/VoidPendingCash/index.php

Method: POST or GET

Field Name	Data Type(Size)	M/O	Description
tranID	n{1..10}	M	PG Transaction ID
amount	n{10,2}	M	The total amount to be paid in one purchase order. 2 decimal points, comma(,) is not allowed.
merchantID	an{1..32}	M	Merchant ID provided by PG
checksum	an{32}	M	This is request integrity protection hash string.

Checksum= md5(tranID . amount . merchantID . verify_key)

Response

PG responds JSON format to merchant upon a successful request

Field Name	Data Type(Size)	M/O	Description
StatCode	n{2}	M	00 = Success (voided) 11 = Missing required parameter (<FieldName>). 12 = Merchant info not found 13 = Invalid checksum value. 14 = Transaction not found 15 = Transaction not Pending 99 = System is busy now, temporary out of services. Please try again later.
tranID	n{1..10}	O	PG Transaction ID
orderid	an{1..32}	O	Merchant order
amount	n{10,2}	O	The total amount to be paid in one purchase order
merchantID	an{1..32}	O	Merchant ID provided by PG
channel	an{1..32}	O	Channel references for merchant system.

Recurring Plans API

For merchants to retrieve recurring plans info which were created from the merchant portal.

Request

URL: https://api.merchant.razer.com/RMS/API/Recurring/get_plans.php

Method: POST or GET

Field Name	Data Type(Size)	M/O	Description
domain	an{1..32}	M	Merchant ID provided by PG
charge_on_endofmonth	a{1}	O	Predefined value as below: 1. Y 2. N
period	a{3..8}	O	Predefined value as below: 1. month 2. week 3. day 4. year 5. quarter 6. halfyear 7. bimonth 8. biweek
cycle_term	n{1..10}	O	Billing cycle (E.g. '6', '12', '24')
status	a{2..3}	O	Predefined value as below: 1. On 2. Off 3. NA
skey	an{32}	M	This is request integrity protection hash string.

Response (JSON format)

Variable / Parameter	Type Format / Max Length	Description / Example
planID	n{1..10}	Unique ID for each plan.
plan_name	ans{1..64}	Plan name.
plan_desc	ans{1..255}	Plan description.
amount	n{10,2}	The total amount to be paid in each cycle.
period	a{3..8}	Predefined value as below: 1. month 2. week 3. day 4. year 5. quarter 6. halfyear 7. bimonth 8. biweek

status	a{2..3}	Predefined value as below: 1. On 2. Off 3. NA
cycle_term	n{1..10}	Billing cycle (E.g. '6', '12', '24')
charge_on_endofmonth	a{1}	Predefined value as below: 1. Y 2. N

PG will respond the following in JSON format once error occurs (negative result)

No	Field Name	Data Type(Size)	Description
1	error_code	an{7}	Refer to Appendix A
2	error_desc	ans{1..255}	Refer to Appendix A

Formula of skey

skey = md5(domain & secret_key & charge_on_endofmonth & period & cycle_term & status)

Appendix A : Error Code & Description in JSON

error_code	error_mesg
PLAN001	Missing required parameter (<FieldName>).
PLAN002	Merchant info not found.
PLAN003	Invalid skey.
PLAN004	Plan not found.
PLAN005	System is busy now, temporary out of services. Please try again later.
PLAN006	Invalid parameter value (<FieldName>).

ERROR CODES

Payment Page

Error Code	Description
P01	Timeout
P02	Your transaction has been denied due to merchant account issue. Payment channel is not available for merchant.
P03	Your payment info format not correct ! Incorrect security hash string, check “vcode”.
P04	not allow to process Incomplete buyer information, check bill_name, bill_mobile, bill_email, orderid, faked mobile such as 0123456789 will be blocked as well
P05	Payment gateway not found Invalid payment gateway file or channel code.
P06	System is busy now, temporary out of services. Please try again later.
P07	Access Denied. Requestor not authorize. Merchant URL is not allowed to process payment or not yet registered with PG.
P08	Invalid referral
P09	Duplicate payment is not allowed for this merchant. Payment with same order ID and amount is not allowed to capture twice. Cash payment is not allowed to request twice.
P10	Sorry, Your Credit Card Number or CVV or expiration date is not valid
P11	Amount return from bank not match with PG system.
P12	Signature from bank not match.
P13	Currency not supported.
P14	Transaction amount must more than CUR XXX.XX.
P15	Transaction amount must less than CUR XXX.XX.
P16	Invalid subMerchantID
P17	Unable to retrieve the currency exchange rate
P18	Empty currency rate
P21	Massive errors detected from the same IP address

P22	Massive errors detected from the same IP address
P33	System is busy
P44	Bill description format incorrect
P45	Subtotal amount not match with bill amount
P46	Merchant is not allowed to use this payment method
P47	Bank under maintenance
P403	Invalid payment URL
P404	Invalid merchant ID
P990	Sorry, we're not able to process your request now. You may close this window.
P991	System temporary not available due to security issue
P1813	URL is not allowed
R01	Error in payment process due to empty key
T01	Tokenization error
01	Unsuccessful Timeout due to user did not authorize the payment
99	Record not found Timeout due to user did not proceed with the payment
999	Transaction is blocked Due to one of the reason: country restriction, blacklist, exceeds transaction limit

3DS Error

Error Code	Description
P3011	This merchant only allows 3DS card
P3089	Invalid 3DS request
P3099	Card not authorized or invalid BIN
MD2310	rejected due to non-3DS (a feature of PG to block non-3DS transaction)

Payment Status Query (PSQ) Error

Error Code	Description
Q01	Missing Required Parameter (*{parameter_name}*).
Q02	Invalid Parameter Value (*{parameter_name}*).
Q03	System is busy now, temporary out of services. Please try again later.
Q04	Merchant info not found.
Q05	Send result to merchant URL failed.
Q06	Query period expired. No result available for transaction more than 7 days.
Q99	Service blocked due to massive connection. Please send only 1 request every 5 minutes.
Q101 / Q201	Correct skey with invalid data
Q00004 / Q102 / Q202 / Q302 / Q402 / Q502	Incorrect skey
Q203 / Q303	Transaction record not found
Q401	Delimiter error

Error in Merchant Admin

Error Code	Description
M0001	Inactive merchant account
M0002	Frozen merchant account
M0003	Merchant account has been purged
M0098	Invalid merchant ID
M0099	Unable to process the request
M1007	Invalid bank account number
M1008	Invalid email
M1009	Invalid bank account number & email

Error Codes from Payment Channel

PG will add the following prefix to all card and certain internet banking error codes

Credit Card (Paymex, MIGS, etc)	CC_
FPX Internet Banking	FPX_

Error Code | Description

Prepaid/Debit/Credit Card via Paymex (Visa/MasterCard)

1XX | Invalid input to 3D Secure MPI
2XX | Error related to 3D authentication
3XX | Error related to authorization
4XX | System error or timeout

Error Code	Description	Action / Remarks
000	Transaction is successful	Merchant to display the confirmation page to cardholder
101	Invalid field passed to 3D Secure MPI	Merchant needs to check error description to find out what is wrong with the field. Authorization/Authentication not carried out.
201	Invalid ACS response format. Transaction is aborted.	Retry the transaction. If error persists, contact the issuing bank.
202	Cardholder failed the 3D authentication, password entered by cardholder is incorrect and transaction is aborted	Merchant to display error page to cardholder
203	3D PaRes has invalid signature. Transaction is aborted	Retry the transaction. If error persists, contact the issuing bank.
300	Transaction not approved	Transaction has failed authorization, e.g. due to insufficient credit, invalid card number, etc. The actual response code provided by acquiring host can be found via the View Transaction History web page available to merchants.
301	Record not found	• Merchant/User has submitted a transaction with invalid purchase ID • Merchant/User tried to reverse a previously declined transaction
302	Transaction not allowed	• Purchase ID not unique due to mismatched card number and/or transaction amount • System unable to process reversal due to transaction has been settled • System unable to process reversal due to transaction type is CAPS • System unable to process previously voided transaction
303	Invalid Merchant ID	Not a valid merchant account

304	Transaction blocked by error 901	Merchant to report error to acquiring bank
305	Merchant password is not available	The error code will be returned if merchant password is not available during the VISA 3D authentication,
306	Merchant exceeded count limit	The error code will be returned if the merchant daily transaction count exceeded the predefined daily transaction count limit.
307	Merchant exceeded amount limit	The error code will be returned if the merchant daily transaction amount exceeded the predefined daily transaction amount limit.
308	Merchant exceeded group count limit	The error code will be returned if the merchant transaction exceeded the predefined merchant organization transaction count limit.
309	Merchant exceeded group amount limit	The error code will be returned if the merchant transaction exceeded the predefined merchant organization transaction amount limit.
310	Request signature is not available	The PX_SIG is not append on the PxReq message or PX_SIG is blank
311	Request signature verification failed	The signature of the PxReq is not match with the PxReq message
900	3D Transaction timeout	Timeout of 3D transaction due to late response from Issuer ACS, after the predefined 3D timeout set in the application.
901	System Error	System unable to complete transaction. Merchant to report error to acquiring bank.
902	Time out	Issuing/acquiring host timeout, transaction is not approved
903	Pxtems no response	Transaction has failed to get response from MPI links. The service not available due to: ● Database connection error ● Network connection error

Host Error
00 | Successful approval/completion or that V.I.P. PIN verification is valid
01 | Refer to card issuer
02 | Refer to card issuer, special condition
03 | Invalid merchant or service provider
04 | Pickup card
05 | Do not honor / Incorrect CVV or 3D password
06 | Error
07 | Pickup card, special condition (other than lost/stolen card)
10 | Partial Approval-Private label
11 | V.I.P. approval
12 | Invalid transaction
13 | Invalid amount (currency conversion field overflow. Visa Cash-invalid load mount)
14 | Invalid account number (no such number)
15 | No such issuer
19 | Re-enter transaction
21 | No action taken (unable to back out prior transaction)
25 | Unable to locate record in file, or account number is missing from the inquiry
28 | File is temporarily unavailable
41 | Pickup card (lost card)
43 | Pickup card (stolen card)
51 | Insufficient funds
52 | No checking account
53 | No savings account
54 | Expired card
55 | Incorrect PIN (Visa Cash-invalid or missing S1 signature)
57 | Transaction not permitted to cardholder (Visa Cash-incorrect routing, not a load request)
58 | Transaction not allowed at terminal

59 | Suspected fraud"; break;
61 | Activity amount limit exceeded
62 | Restricted card (for example, in Country Exclusion table)
63 | Security violation
65 | Activity count limit exceeded
75 | Allowable number of PIN-entry tries exceeded
76 | Unable to locate previous message (no match on Retrieval Reference number)
77 | Previous message located for a repeat or reversal, but repeat or reversal data are in consistent with original message
80 | Invalid date (For use in private label card transactions and check acceptance transactions)
81 | PIN cryptographic error found (error found by VIC security module during PIN decryption)
82 | Incorrect CVV/iCVV
83 | Unable to verify PIN
85 | No reason to decline a request for account number verification or address verification
91 | Issuer unavailable or switch inoperative (STIP not applicable or available for this transaction)
92 | Destination cannot be found for routing
93 | Transaction cannot be completed; violation of law
96 | System malfunction System malfunction or certain field error conditions
N0 | Force STIP
N3 | Cash service not available
N4 | Cash request exceeds issuer limit
N7 | Decline for CVV2 failure
P2 | Invalid biller information
P5 | PIN Change/Unblock request declined
P6 | Unsafe PIN

Card via MIGS (Visa/MasterCard)
0 | Transaction Successful
? | Transaction status is unknown
1 | Unknown Error
2 | Bank Declined Transaction
3 | No Reply from Bank
4 | Expired Card
5 | Insufficient funds
6 | Error Communicating with Bank
7 | Payment Server System Error
8 | Transaction Type Not Supported
9 | Bank declined transaction (Do not contact Bank)
A | Transaction Aborted
C | Transaction Cancelled
D | Deferred transaction has been received and is awaiting processing
F | 3D Secure Authentication failed
I | Card Security Code verification failed
L | Shopping Transaction Locked (Please try the transaction again later)
N | Cardholder is not enrolled in Authentication scheme
P | Transaction has been received by the Payment Adaptor and is being processed
R | Transaction was not processed - Reached limit of retry attempts allowed
S | Duplicate SessionID (OrderInfo)
T | Address Verification Failed
U | Card Security Code Failed
V | Address Verification and Card Security Code Failed
Y | The cardholder was successfully authenticated.
E | The cardholder is not enrolled.
N | The cardholder was not verified.
U | The cardholder's Issuer was unable to authenticate due to some system error at the Issuer.
F | There was an error in the format of the request from the merchant.
A | Authentication of your Merchant ID and Password to the ACS Directory Failed.
D | Error communicating with the Directory Server.
C | The card type is not supported for authentication.
S | The signature on the response received from the Issuer could not be validated.
P | Error parsing input from Issuer.
I | Internal Payment Server system error.

05 | Fully Authenticated
06 | Not authenticated (cardholder not participating), liability shift
07 | Not authenticated due to a system problem

PayNet FPX Internet Banking (MY)
00 | Approved
03 | Invalid Merchant
05 | Seller Account Closed
09 | Transaction Pending
12 | Invalid Transaction
13 | Invalid Amount
14 | Invalid Buyer Account
20 | Invalid Response
30 | Transaction Not Supported For Model Or Format Error
31 | Invalid Bank
39 | No Credit Bank
45 | Duplicate Seller Order Number
46 | Invalid Seller Exchange Or Seller
47 | Invalid Currency
48 | Maximum Transaction Limit Exceeded
49 | Merchant Specific Limit Exceeded
50 | Invalid Seller for Merchant Specific Limit
51 | Insufficient Funds
53 | No Saving Account
57 | Transaction Not Permitted
58 | Transaction to Merchant Not Permitted
65 | Withdrawal Frequency Exceeded
70 | Invalid Serial Number
76 | Transaction Not Found
77 | Invalid Buyer Name Or Buyer Id
78 | Decryption Failed
79 | Host Decline When Down
80 | Buyer Cancel Transaction
83 | Invalid Transaction Model
84 | Invalid Transaction Type
85 | Internal Error At Bank System
87 | Debit Failed Exception Handling
88 | Credit Failed Exception Handling
89 | Transaction Not Received Exception Handling
90 | Bank Internet Banking Unavailable
92 | Invalid Buyer Bank
96 | System Malfunction
98 | MAC Error
99 | Pending for Authorization (Applies to B2B model)
BB | Blocked by Bank
BC | Transaction Cancelled By Customer
DA | Invalid Application Type
DB | Invalid Email Format
DC | Invalid Maximum Frequency
DD | Invalid Frequency Mode
DE | Invalid Expiry Date
DF | Invalid e-Mandate Buyer Bank ID
FE | Internal Error
OE | Transaction Rejected As Not In FPX Operating Hours
OF | Transaction Timeout
SB | Invalid Seller Bank Code
XA | Invalid Source IP Address (Applicable for B2B2 model)
XB | Invalid Seller Exchange IP
XC | Seller Exchange Encryption Error
XE | Invalid Message
XF | Invalid Number of Orders
XI | Invalid Seller Exchange
XM | Invalid FPX Transaction Model
XN | Transaction Rejected Due To Duplicate Seller Exchange Order Number
XO | Duplicate Exchange Order Number
XS | Seller Does Not Belong To Exchange
XT | Invalid Transaction Type
XW | Seller Exchange Date Difference Exceeded
1A | Buyer Session Timeout at Internet Banking Login Page

1B | Buyer failed to provide the necessary info to login to IB Login page
1C | Buyer choose cancel at Login Page
1D | Buyer session timeout at Account Selection Page
1E | Buyer failed to provide the necessary info at Account Selection Page
1F | Buyer choose cancel at Account Selection Page
1G | Buyer session timeout at TAC Request Page
1H | Buyer failed to provide the necessary info at TAC Request Page
1I | Buyer choose cancel at TAC Request Page
1J | Buyer session timeout at Confirmation Page
1K | Buyer failed to provide the necessary info at Confirmation Page
1L | Buyer choose cancel at Confirmation Page
1M | Internet Banking Session Timeout
2A | Transaction Amount Is Lower Than Minimum Limit

Alipay (CN)

Error Code	Description
ILLEGAL_SIGN	Illegal signature
ILLEGAL_ARGUMENT	Illegal parameters
HASH_NO_PRIVILEGE	No sufficient rights to complete the query
ILLEGAL_SERVICE	Service Parameter is incorrect
ILLEGAL_PARTNER	Incorrect Partner ID
ILLEGAL_SIGN_TYPE	sign_type parameter is incorrect
FOREX_MERCHANT_NOT_SUPPORT_THIS_CURRENCY	Cannot support this kind of currency
ILLEGAL_SECURITY_PROFILE	Cannot support this kind of encryption
REPEAT_OUT_TRADE_NO	out_trade_no parameter is repeated
ILLEGAL_CURRENCY	Currency parameter is incorrect
ILLEGAL_PARTNER_EXTERFACE	Service is not activated for this account
SYSTEM_EXCEPTION	Contact Alipay technical support for help
ILLEGAL_TIMEOUT_RULE	Timeout_rule parameter is incorrect
ILLEGAL_CHARSET	Illegal charset
99/TRADE_NOT_EXIST	Customer did not make payment/Fail onboarding

Razer Gold

Error Code	Error Description
40001	Required parameter is required. / Parameter format is invalid.
40002	Invalid API Version.
40003	Invalid Currency Code or not supported.
40004	Duplicate Reference Id. The transaction was refused as a result of a duplicate Reference Id supplied. Currency Code is not match with previous transaction. The transaction was refused as a result of a duplicate Reference Id supplied. Pin is not match with previous transaction.

40005	Invalid Channel Id.
40006	Invalid Amount.
40007	Invalid PIN.
40008	Invalid Client IP Address.
40009	The transaction was declined by Razer Gold because of possible fraudulent activity.
40013	Payment Amount Exceed channel maximum accepted amount.
40014	Payment Amount less than channel minimum accepted amount.
40101	Invalid Application Code.
40102	Unauthorized Server IP Address.
40103	Invalid Signature.
40104	Channel Id not permitted.
40400	Payment not found.

UnionPay (CN)
00 Payment is successful.
01 Refer to card issuer.
03 Invalid merchant.
05 Merchant or seller account closed.
06 Issuer card is expired.
09 Request in progress.
12 Invalid transaction.
13 Invalid amount.
14 Invalid buyer account or card information.
17 Buyer cancel transaction.
22 Suspected malfunction.
25 Failure to search an original transaction.
30 Format error or MAC error.
31 Invalid bank.
39 No credit account.
51 Insufficient fund.
53 No saving account.
57 Transaction not permitted.
61 Withdrawal limit exceeded.
65 Withdrawal frequency exceeded.
76 Transaction not found.
77 Invalid buyer name or buyer Id.
79 Host decline when down.
85 Internal error at bank's system.
87 Debit failed (exception handling).
88 Credit failed (exception handling).
89 Transaction not received (exception handling).
92 Financial institution or intermediate network facility cannot be found for routing.
93 Transaction cannot be completed.
FE Internal error.
OE Transaction rejected as not in operating hours.
OF Transaction timeout.
OA Session timeout.
1A Buyer session timeout at bank login page.
1B Buyer failed to provide the necessary info to login to bank login page.
1C Buyer choose cancel at login page.
1D Buyer session timeout at account selection page.
1E Buyer failed to provide the necessary info at account selection page.

1F Buyer choose cancel at account selection page.
1G Buyer session timeout at TAC request page.
1H Buyer failed to provide the necessary info at TAC request page.
1I Buyer choose cancel at TAC request page.
1J Buyer session timeout at confirmation page.
1K Buyer failed to provide the necessary info at confirmation page.
1L Buyer choose cancel at confirmation page.
2A Transaction amount is lower than minimum limit.

Boost (MY)

Error Code	Description
E0005	Invalid input value mandatory parameter missing
E0006	Message too long. Maximum length characters
E0007	Message too short. Minimum length characters
E0008	Invalid email format
E0009	Invalid date format, expected format
E0010	Invalid Double value format
E0013	Invalid msisdn format
E0016	Error occurred
E0017	Invalid value format for path variable
E0020	Merchant not found
E0021	Merchant integration info not found
E0022	Payment transaction token Expired
E0023	Merchant transaction not found
E0024	Already payment done for given payment Transaction Token
E0027	Customer wallet balance not sufficient
E0032	Validation Failure Invalid Merchant category
E0038	Minimum Bio Data [image count] not found

Grab Pay (MY)

Error Code	Description
99/ Invalid transaction_id	Customer did not make payment
99/Record not found	Record not found
currency_mismatch	Currency used in this request mismatches with merchant configuration.

init_record_not_exist	The initiate payment action is not completed.
payment_not_found	Original payment is not found or not completed for the refund request.
partial_refund_not_allowed	Partial refund is not allowed for this transaction.
merchant_insufficient_balance	Merchant balance is insufficient.
exceed_payment_amount	Refund amount exceeds payment amount
no_record_found	No record found for this transaction
charging	System is processing a charge request
Refunding	System is processing a refund request
invalid_request	The request is missing a required parameter, includes an invalid value, includes a parameter more than once, or is otherwise malformed
unauthorized_client	The client is not authorized to request an authorization code using this method
Access_denied	The resource owner or authorization server denied the request
unsupported_response_type	The authorization server does not support obtaining an authorization code using this method
invalid_scope	The requested scope is invalid, unknown, or malformed
server_error	The authorization server encountered an unexpected condition that prevented it from fulfilling the request
temporarily_unavailable	The authorization server is currently unable to handle the request due to a temporary overloading or maintenance of the server.
interaction_required	The Authorization Server requires some form for end-user interaction to proceed. This error may be returned when the prompt parameter value in the Authentication Request is 'none' but cannot be completed without displaying a user interface for end-user interaction.
login_required	The Authorization Server requires end-user authentication. This error may be returned when the prompt parameter value in Authentication Request is none but cannot be completed without displaying a user interface for end-user authentication
account_selection_required	The end-user is required to select a session at the Authorization server. The end-user may be authenticated at the Authorization server with different associated accounts, but he did not select a session. This error may be returned when the 'prompt' parameter value in Authentication Request is 'none' but cannot be completed without displaying a user interface to prompt for a session to use
consent_required	The Authorization Server requires end-user consent. This error may be returned when the prompt parameter value in the Authentication Request is 'none' but cannot be completed without displaying user interface for end user consent.
invalid_request_uri	The request_uri in the Authorization Request returns an error or contains invalid data
invalid_request_object	The Request parameter contains an invalid Request Object
request_not_supported	The OP does not support the use of the Request parameter
request_uri_not_supported	The OP does not support the use of the request_uri parameter

registration_not_supported	The OP does not support the use of the Registration parameter

MAE by Maybank2U (MY)

Error Code	Description
QR098	Sorry, we could not complete your request at this time. Please try again later
QR090	Mandatory fields missing
QR091	Invalid QR ID
QR092	Invalid Terminal ID
QR122	Your account has been locked or Inactive. Please call our Customer Care Hotline at 1300 888 6688 for assistance
QR155	Invalid Transaction
401	Failed to verify OAuth information

Touch `n Go eWallet (MY)

Error Code	Description	Action/Remarks
00000000/INIT	Order is not paid/Paid but unfinished	
00000000/CLOSED	Order is closed	
PAYING	Order is paid but not finish	Order is paid successfully for PAY-CONFIRM
MERCHANT_ACCEPT	Order is accepted by merchant after order is paid for PAY-CONFIRM	
CANCELLED	Order is cancelled	

WeChat Pay (CN & MY)

Error Code	Description	Action/Remarks
SUCCESS	Payment successful	
REFUND	Order to be refunded	
NOTPAY	Order not paid	
CLOSED	Order closed	
REVOKED	Order revoked	
USERPAYING	Awaiting user to pay	
PAYERROR	Payment failed (payment status failed to be returned by bank or other reasons)	

0000	Successful	
9994	Merchant does not support Institution or Agency mode	The Merchant ID is a direct merchant. Please check the Merchant ID
9995	Sub Merchant ID does not belong to the Merchant	Please check the Sub Merchant ID first
9996	Error Signature Algorithm	Check the sign type and for more information.
9997	Invalid Merchant ID	Invalid Merchant ID or Sub Merchant ID. Please check the ID first
9998	Error verifying signature	Check the signature process in detail
9999	System Error	Call the Query Order API to check the current order status. The Status determined which process will be taken next.
0100	Order does not exist	The order status is unknown. It is suggested to call the API once again
0101	Order does not exist	The order status is unknown. It is suggested to call the API once again
0102	Order does not match	Check the program in detail, and confirm the matching relation between Merchant's Order Number and WeChat Pay Order Number
0203	Order does not match.	Check the program in detail, and confirm the matching relation between Merchant's Order Number and WeChat Pay Order Number.
0001	Payment is in progress	Waiting for the user to enter the password/Pin on the smart phone. The Merchant's backend can check the payment result regularly
0011	Not sufficient funds	The customer's account balance is insufficient and you can suggest the customer to change the account
0012	Reached transaction limit	The current payment mode of the customer has reached transaction limit, and you can suggest the customer to change the way of payment mode
0013	Rejected due to risk assessment	The transaction has been rejected due to risk assessment by WeChat Pay automatically. You can suggest the customer to change the way of payment mode or contact the WeChat Pay Customer Service Team
0014	Reject by bank card issuer	The transaction has been rejected by bank card issuer. You can suggest the customer to change the way of payment mode or contact the card issuer
0015	Invalid QR code	It is suggested that the shop assistant can scan the QR code once again or customer reload the QR code
0201	Order does not exist.	Check the program in detail, and confirm the validity of WeChat Pay Order Number
0201	Order does not exist	Check the program in detail, and confirm the validity of Merchant's Order Number

0202	Order does not match	Check the program in detail, and confirm the matching relation between Merchant's Order Number and WeChat Pay Order Number
0203	Order does not match	Check the program in detail, and confirm the matching relation between Merchant's Order Number and WeChat Pay Order Number
0204	Incorrect order status	The order which has been paid or refunded cannot be closed
0300	Order does not exist	Check the program in detail and confirm the validity of WeChat Pay Order Number
0301	Order does not match	Check the program in detail and confirm the validity of Merchant's Order Number
0302	Order does not match	Check the program in detail and confirm the matching relation between Merchant's Order Number and WeChat Pay Order Number.
0303	Order does not match	Check the program in detail and confirm the matching relation between Merchant's Order Number and WeChat Pay Order Number.
0304	The amount of is not consistent with the original order.	Check the program in detail and confirm the validity of the original order information.
03005	Exceed the refundable amount	The total amount of refunds should not be more than the amount of payment.
03006	Incorrect order status	The unpaid or closed order cannot be refunded
0350	Error verifying Merchant certificate	Check whether the Merchant certificate used is valid
0201	Order does not exist	Check the program in detail, and confirm the validity of Merchant's Order Number.
0203	Order paid for more than 1 day cannot be revoked	Order paid for more than 1 day cannot be revoked and you can call the Refund Order API to submit a refund
0250	Error verifying Merchant certificate	Check whether the Merchant certificated used is valid
0400	Order does not match	Check the program in detail, and confirm the validity of order parameter
0401	Order does not exist	Check the program in detail, and confirm the validity of WeChat Pay Order Number or WeChat Pay Refund Number
0402	Order does not exist	Check the program in detail, and confirm the validity of Merchant's Order Number or Merchant's Refund Number
0500	Failed to download	Confirm the validity of the data parameter first

eNETS Debit

Error Code	Description	Action/Remarks

00000	Payment successful	Transaction is approved.
01000	Payment declined. Call Bank.	Bank has declined the transaction because customer cancelled it at the bank login page.
01001	Payment declined. Please try again.	Bank has declined the transaction with unknown reason and indicated customer to try again.
01002	Payment declined. Please contact card issuer	Customer's account has Insufficient Funds
01003	Payment declined. Please contact Merchant	Bank declined with Security Violation
01005 / 01006	Payment declined. Invalid Account	Bank has declined the transaction because customer does not have a Cheque Account
01007	Payment declined. Invalid Account	Bank has declined the transaction because customer does not have a Savings Account
01010	Payment declined. Exceeded activity/pin-retry Limit	Bank has declined the transaction with unknown reason and indicated customer to try again
01011	Payment declined. Exceeded Account Limit	Bank has declined the transaction because customer exceeded their account limit
01039	Payment declined - User Session Expired	Bank has declined the transaction because of Timeout due to customer inactivity
02001	Payment declined. Time out.	Gateway has declined the transaction because timeout on backend has occurred.
02002	Payment declined. User Session Expired.	Payment declined. User Session Expired.
02003	Payment declined. User Cancelled Txn	Payment declined. User Cancelled Txn
02200	Payment declined. Please contact card issuer	Gateway has declined the transaction because issuer/bank related problems

RESOURCES

Logos of all brand name

Merchant may download the logos from https://merchant.razer.com/v3/media-library/
password: RazerMerchantServices

Mobile SDK/XDK

RMS is now ready to be integrated into your mobile apps. We have released the Mobile SDK and XDK library on GitHub. To apply this, kindly contact support-sa@razer.com and provide your Merchant ID or Company name, Platform (iOS/Android), Apps Name in order to register and authorize 3rd party apps in the PG system.

Supported Shopping Cart

RMS has been integrated with many popular shopping carts, globally. Merchant may refer the complete list on https://github.com/RazerMS
Some payment plugins/add-on/modules can be downloaded from GitHub.

ISO References

http://www.iso.org/iso/country_codes.htm
http://en.wikipedia.org/wiki/ISO_3166-1
http://www.iso.org/iso/currency_codes
http://en.wikipedia.org/wiki/ISO_4217

Handling JSON/PLAIN TEXT using .NET

http://stackoverflow.com/questions/36216464/wcf-webinvoke-which-can-accept-content-type-text-plain
Please use the above custom WebContentTypeMapper if you are using .NET, especially when you encounter this exception message:

> The incoming message has an unexpected message format 'Raw'. The expected message formats for the operation are 'XML'; 'JSON'. This can be because a WebContentTypeMapper has not been configured on the binding. See the documentation of WebContentTypeMapper for more details.