



POSA – API Developer Guide

Version 1.0.1

Table Of Contents

Table Of Contents	2
Version Control	3
Documentation location	4
PayThem VVS System overview	5
Security and encryption	5
Environments	6
Environment endpoints	6
Libraries and examples	6
Accounts and users	7
Public & Private tokens	7
API definition.....	8
Outbound call headers.....	8
Outbound call JSON structure.....	8
Outbound call Parameter explanation.....	9
HTTP POST.....	10
Creating CONTENT field	10
Type handling.....	10
Dates & time	10
Encoding.....	10
Response handling	11
Available API functions summary	12
Call Definition.....	13
Activate	13
Cancel.....	14
Status	15
UAT - User Acceptance Testing procedure	16
Issuing of production credentials.....	17

Version Control

Rev No	Changes	Date
0	Initial draft document developed and submitted for review	2022/11/11
1	Revision, correction and additions. Adding library repository paths	2023/06/15

Documentation location

The API documentation is constantly updated as new requirements or features are added.

The only authorized location for this documentation is:

https://downloads.paythem.net/05_API_Libraries/

If you receive this documentation via email or other means, please immediately check the above link to confirm you have received the latest.

PayThem VVS System overview

The PayThem VVS system is a JSON based, REST-like API service. It enables 3rd parties to consume web services provided by PayThem via the VVS Platform.

3rd party clients can connect to PayThem's Virtual Voucher System (VVS) to expand their current offerings or build new applications via direct integration with PayThem's available warehouse of vouchers or POSA cards. Through this, the client can retain their own identity and branding while expanding their own product offerings within their own application environment, including point of sales activation.

Security and encryption

All communication with the VVS system has multiple layers of encryption, authentication and verification. To secure all transactions, all communications are via a SSL secured URL and must be used with the HTTP POST method (unless the documentation clearly specifies otherwise).

When implementing the API:

- Clients must ensure, when communicating with the VVS system, that the PayThem SSL certificate is valid.
- Only parameters of HTTP POST will be processed. GET / DELETE / PUT are ignored.
- VVS API supports the following encryption protocols:
 - OpenSSL AES-CBC-256-OPENSSL
- **For an extra layer of security, we require each API consuming client to provide us with the public (internet) static IP address of the source server from which API calls will be made. Any calls from any other IP addresses will ignored and could lead the account to be blocked and the IP address blacklisted. DDNS and DHCP IP addresses will not be accepted.**
- The client's server's time zone and a timestamp in format "yyyy-mm-dd H:i:s" format must be inserted into each query to prevent replay attacks. Please ensure your time zone and time is set correctly, preferably by a NTP service to minimize risk. A maximum of 30 second deviation from our servers will be tolerated before errors are raised.
- All character encoding must be UTF-8 based.
- Each client takes full responsibility for the safeguarding of their encryption keys. Production encryption keys will be shared in two parts, to separate client staff.
- Multiple API accounts can be created per client, each with their own encryption keys, static IP addresses and authentication credentials. Each API account will use the customer's global account balance for purchases.
- HTTP compression is enabled on our servers and is preferred / recommended.

Environments

PayThem provides two separate environments for client access.

Environment	Purpose
Demo	<ul style="list-style-type: none">• For testing connectivity between client system and VVS and testing functionality of VVS.• All information is similar to Production environment, but all vouchers issued and requests are fake.• No static IP required.
Production	<ul style="list-style-type: none">• Production environment where all data is real time and all vouchers are real vouchers and POSA requests are honoured and completed.• Each voucher / POSA has a real monetary value and account balances reflect real values.• Client must provide a static, public IP address for additional security.• Client to provide an estimated number of calls to be performed per hour to enable our rate limiter functionality.

Environment endpoints

Environment	URL Endpoint
Demo	https://vvsdemo.paythem.net/API/47067/
Production	https://vvs.paythem.net/API/47067/

When using one of our provided libraries, the library requires two parameter that defines which environment and API service the current call will use.

Libraries and examples

PayThem provides libraries and example implementation for various languages.

Language	Type	git Repository
PHP 7.x	Library	https://bitbucket.org/paythem/php7.x
PHP 8.x	Library	https://bitbucket.org/paythem/inbound_api_php8
PHP 7.3	Library	https://bitbucket.org/paythem/inbound_api_php73
Perl	Example	https://bitbucket.org/paythem/perl
Python 3	Library	https://bitbucket.org/paythem/python3

If a library or example is not available for your environment, please contact PayThem technical support through your distributor representative.

These are generic libraries that work on both the standard and POSA API.

Using the libraries are preferred, as they insert most of the required fields.

Accounts and users

During client subscription process, each client is provided with an account. Within this account, different API users can be created, allowing for different application environments to be created by the client. These API users can either be standard or POSA users, but not both.

For example: API00001 can be used from client server SERVER01 and API00002 can be used from client server SERVER02.

During API calls, each call will require a username and password to be included in the encrypted "CONTENT" parameter, as described below.

Each API user can have different IP whitelisting and max voucher returned requirements.

Public & Private tokens

Each environment has its own, unique public key, private key, username and password combinations.

It is crucial to note that the Demo environment's information will not work on Production environment and repeated posting with incorrect details will lead to the account being locked out and the source IP blacklisted.

During an account's API user creation process, PayThem will provide the client with:

- Public key – Passed unencrypted with each query, base64 encoded.
- Private key – used to encrypt JSON parameters before posting to API server.
- Username – encrypted into each post.
- Password – encrypted into each post.

IMPORTANT: It is the client's responsibility to keep all public, private, username and password details secure and hidden from your end users and non-critical staff. If your credentials are compromised (knowingly or unknowingly) PayThem will not be held liable for any damages.

API definition

Relates to the creation of a valid call to the PayThem API service.

Outbound call headers

Header	Content
X-Public-Key	PUBLIC_KEY as provided by PayThem
X-Hash	HMAC hash generated from the pre-encrypted, JSON encoded string using SHA256 and the PRIVATE_KEY as provided by PayThem
X-Sourceip	Public, static IP of server or firewall

Outbound call JSON structure

A brief overview of the minimum values that are required in the un-encrypted, unencoded JSON that is generated by the libraries and examples. PayThem supplied libraries may also add additional fields.

```
{
  "API_VERSION": "1.0.0", // 1.0.0 is current protocol version
  "SERVER_URI": "", // As per documentation. Libraries will auto-insert this field.
  "SERVER_TIMESTAMP": "yyyy-mm-dd H:i:s", // Server time stamp
  "SERVER_TIMEZONE": "", // Time zone in Region/City format
  "SOURCE_IP": "", // Public, static IP of server or firewall
  "PUBLIC_KEY": "", // Public key as supplied by PayThem
  "USERNAME": "", // Username as supplied by PayThem
  "PASSWORD": "", // Password as supplied by PayThem
  "HASH_STUB": "10 random alpha-numeric", // Required.
  "ENCRYPT_RESPONSE": false, // Enable content encryption of response.
  "FUNCTION": "Function to call",
  "PARAMETERS": [
    // Parameter list as per documentation of current FUNCTION
  ]
}
```

Please note the FUNCTION parameter above. Below the different calls are defined and the correct wording for this parameter is defined.

Outbound call Parameter explanation

Parameter required:

- M = Mandatory
- O = Optional

Auto? = Auto-inserted by libraries.

Field	Req.	Auto?	Description
API_VERSION	M	Y	[string] The version of the protocol that is being used.
SERVER_URI	M	Y	[string] The environment URL as defined in 0.
SERVER_TIMESTAMP	M	Y	[string] Local server time for the time zone the server is in. If this is set, TIME_ZONE also needs to be set. Timestamp must be in "CCYY-MM-DD HH:MM:SS" format.
SERVER_TIMEZONE	M	Y *	[string] Default is set on the PayThem server for each API user. Can be overridden by setting the field in the POST string. Format of: "Region/CityOrCountry", e.g., "Asia/Qatar". Please request the correct TZ specifications from us if you have any doubts. Certain libraries will auto-populate this value.
SOURCE_IP	M		[string] Public, static IP of server or firewall. Will be used to check IP whitelisting on multiple layers.
HASH_STUB	M	Y	[string] Randomly generated alpha-numeric string of minimum 10 characters.
ENCRYPT_RESPONSE	M	Y	[boolean/false] Enable content encryption of response sent from PayThem API service. Response from server will be encrypted with the PRIVATE_KEY of the client.
FUNCTION	M	Y	[string] The requested function, as defined in this document.
USERNAME	M	Y	PayThem supplied username.
PASSWORD	M	Y	Paythem supplied password.
PUBLIC_KEY	M	Y	PayThem supplied public key.
PARAMETERS	O	Y	[array] Default empty array. Named parameter key/value of the function. Documentation per function below.

HTTP POST

The structure of the HTTP POST variables:

Parameter	Description
PUBLIC_KEY	Mandatory. PayThem supplied public key.
CONTENT	Mandatory. Base64 encoded encrypted JSON string as explained in 0.
ZAPI	The Initialization Vector (IV) value used to encrypt the CONTENT when using OpenSSL. This identifies to the server that OpenSSL is being used. IV is a randomly generated, 16 length, alpha-numeric string.
ENC_METHOD	Optional. The encoding type used. Currently ignored. Preparation for version 2.3.

Creating CONTENT field

1. Create array with required parameters.
2. Convert to JSON string.
3. Create HMAC hash (used in headers).
4. Encrypt JSON string.
5. Encode encrypted string to base64.

This functionality is taken care of by the library.

Type handling

Dates & time

During VVS API posts, dates are converted from and to client's time zone automatically.

IMPORTANT: *be sure that your time zone is correctly set and that your server time is not out from international atomic time by more than 29 seconds. Else, determine and pass the time as needed to allow for proper usage.*

IMPORTANT: All dates passed from client to server and server to client will be in the format "CCYY-MM-DD HH:MM:SS".

Encoding

All content must be UTF-8.

Response handling

If no response is returned, there could be a break in communications.

The response (once decoded / decrypted) will contain a JSON string containing the following fields:

- **SERVER_TRANSACTION_ID**
The log ID of the client call. This is used to error check with PayThem support in the event of issues. All calls return a unique call ID.
- **RESULT**
0 = No error, else a code representing an error encountered.
- **ERROR_DESCRIPTION**
A human-legible error description.
- **CONTENT**
A base64 encoded (and optionally encrypted, depending on call parameters) representing a JSON string relevant to the call made.

A PayThem library will base64 decode, decrypt and return a JSON string which can be passed to a JSON handler. Certain libraries, like PHP7.x, will return a JSON object. Please confirm with PayThem support before starting integration.

Available API functions summary

API Function	Description
activate	Activate a voucher card PIN
cancel	Cancel a voucher card that has already been activated
status	Check the status of a voucher card activation

Call Definitions

Activate

Activate a voucher card with specified EAN and serial number.

FUNCTION: activate

Parameter	Value	Type
VC_EAN	EAN13 number for the product	(string) 13
VC_SERIAL	The serial located on the voucher card	(string) 36
CLIENT_REFERENCE	A reference to the transaction for logging	(string) 36

Response

Field	Description	Example
TRANSACTION_ID	The ID of the activation transaction for logging and tracking	(string) 11545540
TRANSACTION_ITEM_ID	The ID of the activation transaction line item for logging and tracking	(string) 18393264
VOUCHER_ID	The ID of the activated voucher card	(string) 19819545
VOUCHER_REFERENCE_ID	Reference of the activation call to the supplier	(string) 2788878
DATETIME	The timestamp of the activation of the voucher card	(string) 2022-11-11 12:00:00
PROVIDER_REFERENCE	Reference passed by the supplier during activation	(string) 636E3D1F63DC
SERIAL	The serial of the voucher card that has been activated	(string) 0100624090576

Cancel

Cancel a voucher card that has previously been activated. An error will be returned if the voucher has not been activated.

FUNCTION: cancel

Parameter	Value	Type
VC_EAN	EAN13 number for the product	(string) 13
VC_SERIAL	The serial located on the voucher card	(string) 36
CLIENT_REFERENCE	A reference to the transaction for logging	(string) 36

Response

Field	Description	Example
ACTIVATION_MSA_ID	Reference of the activation call to the supplier	(string) 2788878
CANCELLATION_MSA_ID	ID of the cancellation activation call to the supplier	(string) 2788879
CANCELLATION_MSA_ITEM_ID	The ID of the cancellation activation line item to the supplier	(string) 14349998
PROVIDER_REFERENCE_ID	Reference passed by the supplier during activation	(string) 636E3D1F63DC
SERIAL	The serial of the voucher card	(string) 0100624090576

Status

Checks the status of a voucher card, if it is unused, has been activated, or has been used

FUNCTION: status

Parameter	Value	Type
VC_EAN	EAN13 number for the product	(string) 13
VC_SERIAL	The serial located on the voucher card	(string) 36

Response

Field	Description	Example
LAST_MSA_TRANSACTION	ID of the status check call to the supplier	(string) 2788883
LAST_MSA_TRANSACTION_ITEM	ID of the status check call item to the supplier	(string) 14350007
SERIAL	The serial of the voucher card	(string) 0100624090576
CODE	The voucher status code from the supplier	(string) 42751
EAN	The barcode EAN13 of the voucher card	(string) 4250139876499
CURRENCY	The currency of the voucher card	(string) EUR
PRODUCT_ID	The system ID of the product for the voucher card	(string) 8102
DESCRIPTION	A description for the product	(string) Testprodukt 10 EUR POSA
VALUE	The face value of the voucher card	(string) 10.00
CAN_CANCEL	If the voucher can be cancelled	(bool) true
STATUS	The status of the voucher card as readable text	(string) ACTIVATED

UAT - User Acceptance Testing procedure

Before a client can be issued production credentials, an acceptance test by PayThem will be conducted to assure that all required features have been integrated.

Minimum required calls to proceed to UAT phase:

- Activate
- Cancel
- Status

Once all these calls have been confirmed to be processed successfully, production credentials will be issued.

Issuing of production credentials

Once UAT is complete, only your designated Distributor administrator can request.

Please contact them directly and they will submit the relevant tickets to the correct department.

Credentials will be shared in two parts:

- Public key / username – via email.
- Private key / password – any agreed, non-email communication method.

For production environment, a static IP address is required. No user will be created unless a static IP is provided.
