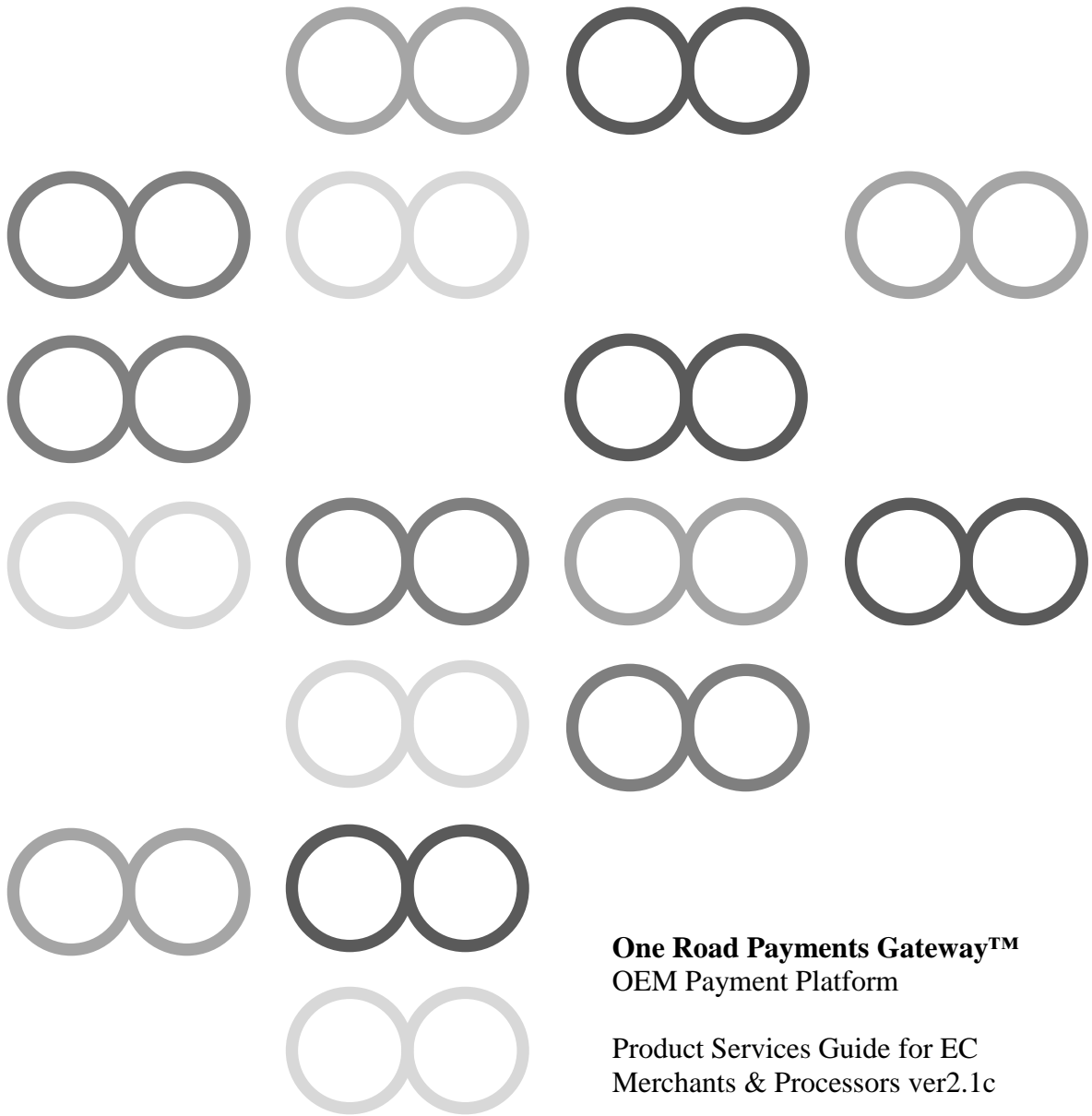One Road Payments Gateway™ Product Services Guide for EC Merchants and Processors

One Road Payments Gateway™
OEM Payment Platform

Product Services Guide for EC
Merchants & Processors ver2.1c

May 2018
One Road Payments © 2017, 2018

Important Notices
One Road Payments owns the intellectual property in this document exclusively. You acknowledge that you must not perform any act which infringes the copyright or any other intellectual property rights of One Road Payments and cannot make any copies of this Manual unless in accordance with these terms and conditions.

Without our express written consent, you must not:

• Distribute any information contained in this Manual to the public media or quote or use such information in the public media; or

• Allow access to the information in this Manual to any company, firm, partnership, association, individual, group of individuals or other legal entity other than your officers, directors and employees who require the information for purposes directly related to your business.

The software described in this Manual is supplied under a license agreement and may only be used in accordance with the terms of that agreement.

One Road Payments, One Road Payments Gateway and the One Road Payments logo are trademarks of One Road Payments.

All third-party product and service names are trademarks or registered trademarks of their respective owners.

1. OVERVIEW

One Road Payments has implemented a Payment Gateway facility to enable merchants, payment processors and payment service providers to accept common payment cards for Electronic Commerce (EC) and Point of Sale (POS). This document is specific to EC merchants only.

The service is known as **One Road Payments Gateway**, and provides internet front-end payment and account management functions and an on-line administrative platform to a web hosted application.

The purpose of One Road Payments Gateway is to provide a high function payment service that simplifies the payment acceptance process for merchants.

One Road Payments Gateway provides an outsourced proposition to acquirers, merchants and payment processors (collective referred to as *Payment Service Providers* or PSPs in this document) for this type of service providing:

Significant cost savings – costs incurred for redundant hardware, software, telecommunications and internal operational overheads for an in-house system are avoided.

Faster time to market – One Road Payments Gateway makes it possible for a user to be "live" with their service within a much shorter time frame than usual.

Reduced risk – One Road Payments Gateway clients avoid many of the risks associated with an in-house system such as high initial and on-going costs; recruitment of skilled technical resources; data security requirements and ability to say up to date with the latest industry requirements, compliance and trends.

Business transparency – One Road Payments Gateway is an "out-sourced" packaged service running on Amazon Web Services. All costs are clear to the merchant or processor and agreed up front.

Reduced burden of merchant administration - One Road Payments Gateway provides a web-based merchant administration portal for merchants and processors to manage their payment transactions. The system allows merchants to report on transaction types and perform financial transactions on-line such as refunds, voids and reconciliation.

Wide applicability - One Road Payments Gateway can be used to enable individual merchants regardless of size and number of transactions. It is an ideal solution for the small to medium enterprise (SME) but can also be readily tailored to suit larger enterprises that operate call centres and web sites to service their clients.

Rapid merchant payment enablement - One Road Payments Gateway deploys a flexible connection mechanism based on 'web-services' communication.

Range of payment methods - One Road Payments Gateway currently supports credit and debit card payments from UnionPay, Visa and Mastercard cards. Transaction types include Purchases, Void and Refunds.

Secure payments - Data passed between the merchant and One Road Payments Gateway is encrypted using industry standard cryptography. One Road Payments Gateway can reduce the transaction risk by storing card details and other sensitive data on behalf of the merchant.

Future proofing - One Road Payments Gateway will continue to be compliant with new industry initiatives.

2. PRODUCT DESCRIPTION

2.1.Services Provided to the Merchants

The services provided to merchants and processors by One Road Payments Gateway through the hosted platform are functionally divided into the following groups:

1. Client Account Management
2. Payment Enablement and Processing
3. Merchant Administration Functions

The *Client Account Management* and *Payment Enablement and Processing* services are accessible to the merchant and processors site (for EC merchants) using a Web-Services style connection mechanism, where messages are exchanged using JSON encoded resources via RESTful APIs over an encrypted channel. This ensures a simpler integration and allows One Road Payments Gateway to support several integration environments, such as Microsoft COM and .NET, JSP and PHP, etc.

The *Merchant Administration* functions are supplied by providing the merchant or processor with secure browser access to the One Road Payments Gateway central transaction database to perform day-to-day merchant functions. The user is provided with tools for searching, reporting and editing of client accounts and transactions. The user documentation for the Merchant Administration functions is in a separate document, the *One Road Payments Gateway User Interface Guide*.

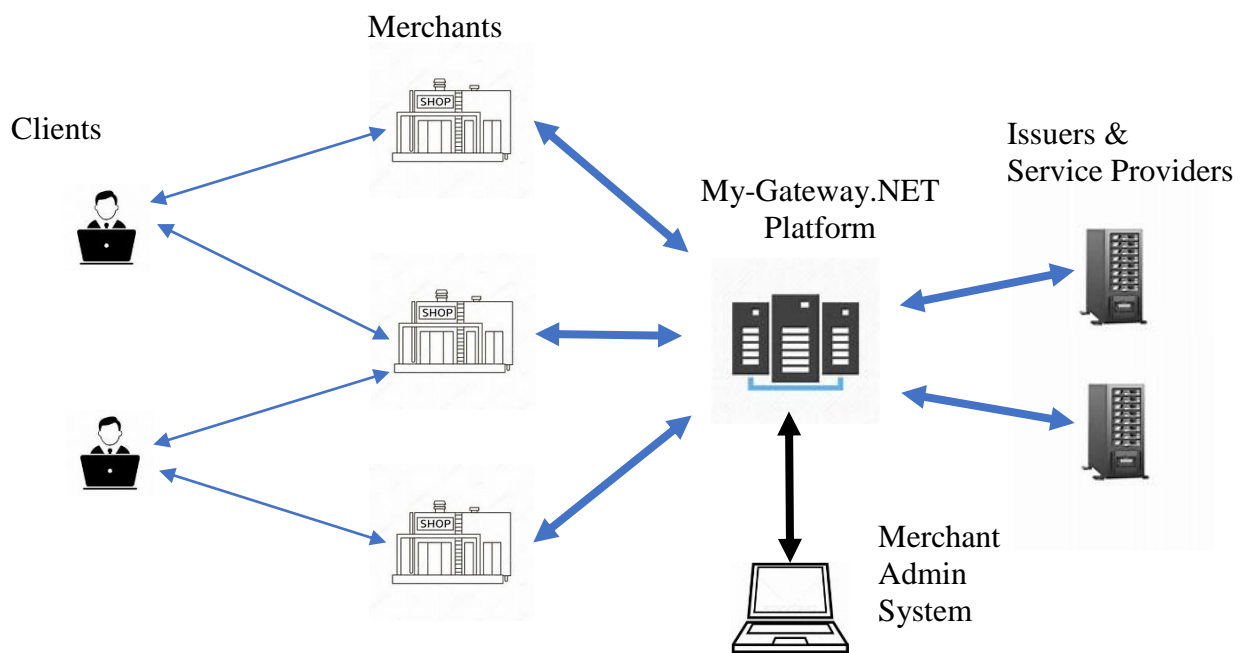The following diagram describes the One Road Payments Gateway system.

Figure 1. One Road Payments Gateway Architecture

2.2. Settlement

Settlement is the process whereby funds are transferred to merchants' bank accounts from their PSP. Merchant funds are transferred into their bank account after their transactions have been authorised by the card issuing institution and funds have been cleared by the relevant card scheme network. Settlement usually takes place a number of days after the transactions have been approved, the 'Settlement Date'. The settlement date is generally agreed between the merchant and their PSP prior to enablement.

Currency

One Road Payments Gateway supports any currency for transactions and supports PSPs to settle in the currency of their choice. However, the *One Road Payments Gateway Merchant Administration, ORPG-MA* functions are shown in the merchants' transaction currency.

Refunds

For any goods purchased with a card that are accepted for return, or for any services that are terminated or cancelled, or where any price adjustment is made, merchants may need to refund the transaction amount to the cardholder's account. Card schemes require a refund to only be processed on the same card number that was used in the original sales transaction and for an amount which is no more than the original transaction.

In the majority of cases, merchants and PSPs are only able to perform refund transactions using the *Merchant Administration System, ORPG-MA* described in the relevant user interface document.

Cancel/Void

Merchants and PSPs are able to cancel or void a transaction which has been authorised by the card issuer prior to settlement. A transaction may need to be cancelled if the cardholder completes a payment transaction in error, for the wrong amount or would prefer to choose a different method of payment. Cancels can only be processed on previously authorised transactions which have not yet been cleared by the card issuers and a cancellation will be performed for the entire transaction amount.

Cancel or Void transactions can be performed using the *Merchant Administration System, ORPG-MA* described in the relevant user interface document or via an API call from the merchant or processor site described in section 4 of this document, *Payment Enablement and Processing, ORPG-PM*.

3. CLIENT ACCOUNT MANAGEMENT (ORPG-AM)

The Client Account Management System or One Road Payments Gateway Account Management system (ORPG-AM), is designed for merchants and PSPs who need to manage an on-going relationship with their clients. While merchants and PSPs may provide an account/client management system hosted on their own servers; ORPG-AM provides this service, freeing merchants and PSPs from the need to manage sensitive client data (including potentially payment card information).

ORPG-AM provides each merchant and processor with a database hosted on the One Road Payments Gateway platform. Client information is stored on the merchant's ORPG-AM database and accessed via Interface Calls from the merchant or processor website.

3.1. ORPG-AM Interface Calls

The following interface calls are available to the merchant or processor from their website to access the ORPG-AM functions:

1. **Add Client** – used to add a client's details to the merchant's specific ORPG-AM database.
2. **Get Client Information** – used to return a specific client's details stored in the ORPG-AM database including payment transaction history.
3. **Update Client Information** – used to update existing client information or add new client information to an existing client entry in the ORPG-AM database.
4. **Delete Client** – used to remove a client's details from the ORPG-AM database.
5. **List Clients** – used to retrieve a list of all the clients in the ORPG-AM database for a specific merchant.

Note that all these functions can also be manually performed by the merchant and PSP using the merchant administration system described in the One Road Payments Gateway Merchant Administration User Interface Guide.

Calls are initiated through RESTful APIs to the ORPG-AM system. The URL used to address each of the API calls includes the unique One Road Payments Gateway identifier assigned to the merchant.

The payload for all client related functions is a ***client*** Object, serialised as a JSON string. A One Road Payments Gateway client Object holds the details associated with an individual user of the payment system. This user has an association with a given merchant and has been registered as a client of said merchant.

The JSON object is detailed below:

Example of JSON client object:

```
{
        "id": "6721f4f7-8012-43aa-a7d0-68458a2f9e16",
        "clientID": "client-001",
        "merchantID": "8873abde-7b40-4ac2-8774-be331fa80ffd",
        "merchantName": "Joe's Online Shop",
        "mobileNumber": "59472680",
        "created": "2017-10-05T02:23:52.645Z",
        "updated": "2017-10-17T21:36:50.575Z",
        "emailAddress": "joe@onlineshop.com.hk",
        "landlineNumber": "",
        "billingAddress": {
            "streetNumber": "23",
            "streetName": "Tai Po",
            "streetSuffix": "Rd",
            "city": "Cheung Sha Wan",
            "state": "",
            "postCode": "",
            "country": "Hong Kong",
            "valid": true
        },
        "cards": [
            {
                "pan": "6223164891230064",
                "expiry": "12/33",
```

```
              "name": "Joe F Bloggs",
              "expiryMonth": "12",
              "expiryYear": "2033",
              "default": true,
              "valid": true;
        },
        {
              "pan": "6250947000000017",
              "expiry": "12/33",
              "name": "Mary J Bloggs",
              "expiryMonth": "12",
              "expiryYear": "1967",
              "default": false,
              "valid": true
        }
    ],
    "valid": true
}
```

Color Key:
Blue: *System supplied, read-only.*
Orange: *Mandatory.*
Normal: *Optional.*

Parameters

- *id*   unique client identifier, supplied by One Road Payments Gateway.

- *clientID*   client identifier, supplied by merchant.

- *merchantID*   unique merchant identifier, supplied by One Road Payments Gateway. This is the identifier for the merchant, who has registered the client.

- *merchantName*   merchant trading name.

- *mobileNumber*   mobile/cell telephone number.

- *created*   date and time when the client record was created, in the One Road Payments Gateway system.

- *updated*   date and time when the client record was last updated, in the One Road Payments Gateway system.

- *emailAddress*   email address of the client.

- *landlineNumber*   contact landline telephone number for the client. This number is not used by One Road Payments Gateway, but is stored to assist the merchant.

- *billingAddress*   billing address of the client. This may be used by credit card authorization system as an authentication mechanism.

- *streetNumber*   the number on a street of the client's postal

address.

- *streetName*   the name of the street of the client's postal address.

- *streetSuffix*   the tail of the street address, of the client's postal address, typically St, Rd, etc.

- *city*   the town, city or village that is used in the client's postal address.

- *state*   the state or region used in the client's postal address.

- *postcode*   the postal code used in the client's postal address.

- *country*   the country used in the client's postal address.

- *cards*   an array of registered credit cards for the client. There must be at least one card registered. Each card will have the fields listed below.

- *pan*   the Primary Account Number, or that embossed on the front of the credit card.

- *expiry*   the expiration date of the credit card.

- *name*   the name of the person that the credit card was issued to.

- *expiryMonth*   this is a derived field, and it the month of the card expiration date.

- *expiryYear*   this is a derived field, and it the year of the card expiration date.

- *default*   This field indicates that this is the default card, of the cards listed in the array

The following sections provide details on the ORPG-AM interface functions.

Add Client

This call is used to add a client's details to the merchant's specific ORPG-AM database.

URLs:
https://api. test.oneroadpayments.com/merchants/{*merchantID*}/clients
https://api.oneroadpayments.com/merchants/{*merchantID*}/clients

Request Method:
POST

URI Parameters:
Required:

- *merchantID* uniquely identifies a merchant within the One Road Payments Gateway system.

Query Parameters:

- *None*

Body Parameters:
The *Body* transfers all the information to be associated with the new client, formatted as a *client* JSON object referred to on Page 10.

The following parameters are mandatory for this request:

- *clientID* client identifier, supplied by merchant.

- *mobileNumber* mobile/cell telephone number.

- *emailAddress* email address of the client.

Success Response:
A successful request will receive a 201 response, along with an updated *client* object.

201	Successfully Created.

Error Response:
An unsuccessful request will receive one of the following responses, along with a message indicating the reason for the failure.

400	Invalid merchant ID.
	Malformed request, missing path parameter or body.
409	ClientID already exists, for merchantID: %s
500	Internal processing exception.

Sample Code:

```html
<!doctype html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge, chrome=1">
    <meta name="description" content="One Road Payments Gateway">
    <meta name="author" content="One Road Payments">
    <title>One Road Payments Gateway</title>
    <base href="./">
    <style type="text/css">
        .container {
            width: 300px;
            clear: both;
        }
        .container input {
            width: 100%;
            clear: both;
        }
    </style>
    <script>
        function UserAction() {
            var xhttp = new XMLHttpRequest();
            var merchantID = document.getElementById("merchantID").value;
            xhttp.open("POST", "https://api.test.oneroadpayments.com/merchants/" + merchantID
+ "/clients", false);
            xhttp.setRequestHeader("content-type", "application/json");
            xhttp.setRequestHeader("x-api-key", "i74wnDTxoi4Nxy61aatbr6mNj2KQtDb31iqrQJ1c");
            xhttp.send('{"clientID":"' + document.getElementById("clientID").value +
                '","emailAddress":"' + document.getElementById("email").value +
                '","mobileNumber":"' + document.getElementById("mobile").value + '"}');
            document.getElementById("result").innerText =
JSON.stringify(JSON.parse(xhttp.responseText), undefined, 2);
        }
    </script>
</head>
<body>
    <div class="container">
        <form>
            <div class="form-group">
                <label for="merchantID">MerchantID</label>
                <input type="text" autofocus placeholder="Merchant Identifier" id="merchantID"
name="merchantID" value="8873abde-7b40-4ac2-8774-be331fa80ffd">
            </div>
            <div class="form-group">
                <label for="clientID">ClientID</label>
                <input type="text" placeholder="Client Identifier" id="clientID"
name="clientID" value="">
            </div>
            <div class="form-group">
                <label for="mobile">Mobile Number</label>
                <input type="text" placeholder="Mobile Telephone#" id="mobile" name="mobile"
value="">
            </div>
            <div class="form-group">
                <label for="email">Email</label>
                <input type="text" placeholder="Email Address" id="email" name="email"
value="">
            </div>
            <br>
            <br>
            <div class="form-group">
                <pre id="result"></pre>
            </div>
            <button type="button" onclick="UserAction()">Add Client</button>
        </form>
    </div>
</body>

</html>
```

Notes:

The sample HTML/Javascript code, above, is a minimalist implementation to demonstrate how easy it is to add a client to the One Road Payments Gateway database. Many more fields should be added to make this a meaningful operation, for example there is no card data which would stop any payments being made by this client.

Get Client Details

This call is used to return a specific client's details stored in the ORPG-AM database including payment transaction history.

URLs:
https://api.test.oneroadpayments.com/merchants/{*merchantID*}/clients/{clientID}
https://api.oneroadpayments.com/merchants/{*merchantID*}/clients/{clientID}

Request Method:
GET

URI Parameters:
Required:

- *merchantID*   uniquely identifies a merchant within the One Road Payments Gateway system.

- *clientID*   uniquely identifies a client within a merchant's registered client collection. This can be either the *clientID* or *id* parameters detailed in the *client* object above, both are supported.

Query Parameters:
- *None*

Body Parameters:
- *None*

Success Response:
A successful request will receive a 200 response, along with the requested *client* object.

     200   OK.

Error Response:
An unsuccessful request will receive one of the following responses, along with a message indicating the reason for the failure.

     400   Malformed request, missing path parameters.

     404   No Client found with MerchantID: %s and ClientID: %s.

     500   Internal processing exception.

Sample Code:

```
<!doctype html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge, chrome=1">
    <meta name="description" content="One Road Payments Gateway">
    <meta name="author" content="One Road Payments">

    <title>One Road Payments Gateway</title>
    <base href="./">
    <style type="text/css">
        .container {
```

```
                width: 300px;
                clear: both;
        }

        .container input {
                width: 100%;
                clear: both;
        }
    </style>
    <script>
        function UserAction() {
                var xhttp = new XMLHttpRequest();
                var merchantID = document.getElementById("merchantID").value;
                var clientID = document.getElementById("clientID").value;
                xhttp.open("GET", "https://api.test.oneroadpayments.com/merchants/" + merchantID +
"/clients/" + clientID, false);
                xhttp.setRequestHeader("x-api-key", "i74wnDTxoi4Nxy61aatbr6mNj2KQtDb31iqrQJ1c");
                xhttp.send();
                document.getElementById("result").innerText =
JSON.stringify(JSON.parse(xhttp.responseText), undefined, 2);
        }
    </script>
</head>

<body>
    <div class="container">
        <form>
            <div class="form-group">
                <label for="merchantID">MerchantID</label>
                <input type="text" autofocus placeholder="Merchant Identifier" id="merchantID"
name="merchantID" value="8873abde-7b40-4ac2-8774-be331fa80ffd">
            </div>
            <div class="form-group">
                <label for="clientID">ClientID</label>
                <input type="text" placeholder="Client Identifier" id="clientID"
name="clientID" value="">
            </div>
            <br>
            <br>
            <div class="form-group">
                <pre id="result"></pre>
            </div>
            <button type="button" onclick="UserAction()">Get Client</button>
        </form>
    </div>
</body>

</html>
```

Notes:

The sample HTML/Javascript code, above, is a minimalist implementation to demonstrate how easy it is to retrieve a client's details from the One Road Payments Gateway database.

Update Client Information

This call is used to update existing client information or add new client information to an existing client entry in the ORPG-AM database.

URLs:
https://api.test.oneroadpayments.com/merchants/{*merchantID*}/clients/{clientID}
https://api.oneroadpayments.com/merchants/{*merchantID*}/clients/{clientID}

Request Method:
PUT

URI Parameters:
Required:

- *merchantID* uniquely identifies a merchant within the One Road Payments Gateway system.

- *clientID* uniquely identifies a client within a merchant's registered client collection. This can be either the *clientID* or *id* parameters detailed in the *client* object above, both are supported.

Query Parameters:

- *None*

Body Parameters:
The *Body* transfers all the information to be updated for the associated client, formatted as a *client* JSON object referred to on Page 10. It is mandatory that a *client* Object be included in the request for Update.

Success Response:
A successful request will receive a 200 response, along with an updated *client* object.

200	OK.

Error Response:
An unsuccessful request will receive one of the following responses, along with a message indicating the reason for the failure.

400	Malformed request, missing path parameters, or Body.
404	No Client found with MerchantID: %s and ClientID: %s.
500	Internal processing exception.

Sample Code:

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge, chrome=1">
    <meta name="description" content="One Road Payments Gateway">
    <meta name="author" content="One Road Payments">
    <title>One Road Payments Gateway</title>
```

```
    <base href="./">
    <style type="text/css">
        .container {
            width: 300px;
            clear: both;
        }
        .container input {
            width: 100%;
            clear: both;
        }
    </style>
    <script>
        function UserAction() {
            var xhttp = new XMLHttpRequest();
            var merchantID = document.getElementById("merchantID").value;
            var clientID = document.getElementById("clientID").value;
            xhttp.open("PUT", "https://api.test.oneroadpayments.com/merchants/" + merchantID +
"/clients/" + clientID, false);
            xhttp.setRequestHeader("content-type", "application/json");
            xhttp.setRequestHeader("x-api-key", "i74wnDTxoi4Nxy61aatbr6mNj2KQtDb31iqrQJ1c");
            xhttp.send('{"clientID":"' + document.getElementById("clientID").value +
                '","emailAddress":"' + document.getElementById("email").value +
                '","mobileNumber":"' + document.getElementById("mobile").value + '"}');
            document.getElementById("result").innerText =
JSON.stringify(JSON.parse(xhttp.responseText), undefined, 2);
        }
    </script>
</head>
<body>
    <div class="container">
        <form>
            <div class="form-group">
                <label for="merchantID">MerchantID</label>
                <input type="text" autofocus placeholder="Merchant Identifier" id="merchantID"
name="merchantID" value="8873abde-7b40-4ac2-8774-be331fa80ffd">
            </div>
            <div class="form-group">
                <label for="clientID">ClientID</label>
                <input type="text" placeholder="Client Identifier" id="clientID"
name="clientID" value="">
            </div>
            <div class="form-group">
                <label for="mobile">Mobile Number</label>
                <input type="text" placeholder="Mobile Telephone#" id="mobile" name="mobile"
value="">
            </div>
            <div class="form-group">
                <label for="email">Email</label>
                <input type="text" placeholder="Email Address" id="email" name="email"
value="">
            </div>
            <br>
            <br>
            <div class="form-group">
                <pre id="result"></pre>
            </div>
            <button type="button" onclick="UserAction()">Add Client</button>
        </form>
    </div>
</body>
</html>
```

Notes:

The sample HTML/Javascript code, above, is a minimalist implementation to demonstrate how easy it is to update a client's details from the One Road Payments Gateway database.

Delete Client

This call is used to remove a client's details from the ORPG-AM database.

URLs:
https://api.test.oneroadpayments.com/merchants/{*merchantID*}/clients/{clientID}
https://api.oneroadpayments.com/merchants/{*merchantID*}/clients/{clientID}

Request Method:
DELETE

URI Parameters:
Required:
- *merchantID*   uniquely identifies a merchant within the One Road Payments Gateway system.

- *clientID*   uniquely identifies a client within a merchant's registered client collection. This can be either the *clientID* or *id* parameters detailed in the *client* object above, both are supported.

Query Parameters:
- *None*

Body Parameters:
- *None*

Success Response:
A successful request will receive a 200 response, along with the *client* object that has just been deleted.

200	OK.

Error Response:
An unsuccessful request will receive one of the following responses, along with a message indicating the reason for the failure.

400	Malformed request, missing path parameters.
404	No Client found with MerchantID: %s and ClientID: %s.
500	Internal processing exception.

Sample Code:

```html
<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge, chrome=1">
    <meta name="description" content="One Road Payments Gateway">
    <meta name="author" content="One Road Payments">
    <title>One Road Payments Gateway</title>
    <base href="./">
    <style type="text/css">
        .container {
            width: 300px;
            clear: both;
        }
        .container input {
            width: 100%;
            clear: both;
        }
    </style>
    <script>
        function UserAction() {
            var xhttp = new XMLHttpRequest();
            var merchantID = document.getElementById("merchantID").value;
            var clientID = document.getElementById("clientID").value;
            xhttp.open("DELETE", "https://api.test.oneroadpayments.com/merchants/" +
merchantID + "/clients/" + clientID, false);
            xhttp.setRequestHeader("content-type", "application/json");
            xhttp.setRequestHeader("x-api-key", "i74wnDTxoi4Nxy61aatbr6mNj2KQtDb31iqrQJ1c");
            xhttp.send();
            document.getElementById("result").innerText =
JSON.stringify(JSON.parse(xhttp.responseText), undefined, 2);
        }
    </script>
</head>
<body>
    <div class="container">
        <form>
            <div class="form-group">
                <label for="merchantID">MerchantID</label>
                <input type="text" autofocus placeholder="Merchant Identifier" id="merchantID"
name="merchantID" value="8873abde-7b40-4ac2-8774-be331fa80ffd">
            </div>
            <div class="form-group">
                <label for="clientID">ClientID</label>
                <input type="text" placeholder="Client Identifier" id="clientID"
name="clientID" value="">
            </div>
            <br>
            <br>
            <div class="form-group">
                <pre id="result"></pre>
            </div>
            <button type="button" onclick="UserAction()">Add Client</button>
        </form>
    </div>
</body>
</html>
```

Notes:

The sample HTML/Javascript code, above, is a minimalist implementation to demonstrate how easy it is to delete a client's details from the One Road Payments Gateway database.

List Clients

This call is used to retrieve a list of all the clients in the ORPG-AM database for a specific merchant.

URLs:
https://api.test.oneroadpayments.com/merchants/{*merchantID*}/clients
https://api.oneroadpayments.com/merchants/{*merchantID*}/clients

Request Method:
GET

URI Parameters:
Required:
- *merchantID*   uniquely identifies a merchant within the One Road Payments Gateway system.

Query Parameters:
- *None*

Body Parameters:
- *None*

Success Response:
A successful request will receive a 200 response, along with an Json array of *client* objects.

  200   OK.

Error Response:
An unsuccessful request will receive one of the following responses, along with a message indicating the reason for the failure.

  400   Malformed request, missing path parameters.

  500   Internal processing exception.

Sample Code:

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge, chrome=1">
    <meta name="description" content="One Road Payments Gateway">
    <meta name="author" content="One Road Payments">
    <title>One Road Payments Gateway</title>
    <base href="./">
    <style type="text/css">
        .container {
            width: 300px;
            clear: both;
        }
        .container input {
            width: 100%;
            clear: both;
        }
    </style>
    <script>
        function UserAction() {
            var xhttp = new XMLHttpRequest();
            var merchantID = document.getElementById("merchantID").value;
```

```
xhttp.open("GET", "https://api.test.oneroadpayments.com/merchants/" + merchantID + "/clients",
false);
            xhttp.setRequestHeader("Accept", "application/json");
            xhttp.setRequestHeader("x-api-key", "i74wnDTxoi4Nxy61aatbr6mNj2KQtDb31iqrQJ1c");
            xhttp.send();
            document.getElementById("result").innerText =
JSON.stringify(JSON.parse(xhttp.responseText), undefined, 2);
            }
    </script>
</head>
<body>
    <div class="container">
        <form>
            <div class="form-group">
                <label for="merchantID">MerchantID</label>
                <input type="text" autofocus placeholder="Merchant Identifier" id="merchantID"
name="merchantID" value="8873abde-7b40-4ac2-8774-be331fa80ffd">
            </div>
            <br>
            <br>
            <div class="form-group">
                <pre id="result"></pre>
            </div>
            <button type="button" onclick="UserAction()">Add Client</button>
        </form>
    </div>
</body>
</html>
```

Notes:

The sample HTML/Javascript code, above, is a minimalist implementation to demonstrate how easy it is to list all clients for a given merchant from the One Road Payments Gateway database.

4. PAYMENT ENABLEMENT AND PROCESSING (ORPG-PM)

The Payment Enablement and Processing or One Road Payments Gateway Payment Management system (ORPG-PM) uses a Server Hosted architecture for EC merchants where the Client can only input their card details directly via a web page that is displayed from the ORPG-PM web server.

ORPG-PM transactions use the SSL protocol to provide secure transmission of sensitive data between a client's web browser and the ORPG-PM system. The merchant or PSP does not need to take responsibility for securing card details and only needs to provide ORPG-PM with the payment amount and Client Identifier for any transaction.

4.1.ORPG-PM Interface Calls

The following interface calls are available to the merchant or processor from their website to access the ORPG-PM functions:

1. **Purchase** – used to initiate a payment process to allow a client to pay for goods and services provided by the merchant using their debit and credit cards.
2. **Query** – used to validate the status of a previously executed Purchase.
3. **Void/Purchase Cancellation*** – used to cancel a purchase made prior to the to the merchant being settled for the purchase.
4. **Refund*** – used to refund money to a client's card after the merchant has been settled for the purchase.

*Note that the Void and Refund transactions can also be manually performed by the merchant using the merchant administration system described the One Road Payments Gateway User Interface Guide.

While the Purchase transaction results in the client being directed to a web page served from the ORPG-PM platform, the Void and Refund transactions are processed by the system without client input.

Whether a client has been registered on the ORPG-PM system or not by the merchant, the typical steps in a ORPG-PM payment transaction are as follows:

For UnionPay, Visa and Mastercard Cards

1. The client selects a commodity or service on the merchant's website. The merchant then initiates a request to the ORPG-PM system using a POST request and including the client ID (if available) and payment amount.

2. ORPG-PM confirms the merchant and client identification and starts the payment process.

3. ORPG-PM navigates the cardholder's browser to the payment page. On this page, the client enters payment information including card number, password and expiry date. In addition, clients using UnionPay cards may input their cell phone number and perform an SMS verification process.

4. ORPG-PM then packages the payment information and sends a payment request message to the card issuer.

5. The issuer processes the transaction and responds to ORPG-PM after completing authorization.

6. ORPG-PM receives the transaction result from the issuer and informs the client of the result of the transaction.

7. ORPG-PM then sends the transaction result to the merchant. The merchant can then update its order information and provide the relevant goods or services to the client.

For WeChat Pay and Alipay Mobile Applications

1. The client selects a commodity or service on the merchant's website. The merchant then initiates a request to the ORPG-PM system using a POST request and including the client ID (if available) and payment amount.

2. ORPG-PM confirms the merchant and client identification and starts the payment process.

3. ORPG-PM navigates the cardholder's browser to the payment page. On this page, the client is presented with a unique QR Code and information on the payment amount and payment currency.

Figure 2. Example of a QR Code

4. The client then scans the QR Code using their WeChat or Alipay mobile application payment function, the amount of the transaction is then presented to the client who is asked to approve the transaction and enter their payment password on the WeChat or Alipay Application.

5. The payment information is sent in a payment request message to the card issuer.

5. The issuer processes the transaction and if approved a 'Success' message is displayed on the client WeChat or Alipay Application with the amount and a response is sent to ORPG-PM after completing authorization.

6. ORPG-PM receives the transaction result from the issuer and informs the client of the result of the transaction.

7. ORPG-PM then sends the transaction result to the merchant. The merchant can then update its order information and provide the relevant goods or services to the client.

ORPG-PM interface calls are initiated through a POST or other resource method, with the associated data provided as described in the following sections.

Note that One Road Payments will only provide the information required to initiate these calls to licensed merchants or merchants' developers as required.

Purchase

(If the supporting information for this call is not supplied below, merchants should contact One Road Payments)

This function is used to initiate a payment process to allow a client to pay for goods and services provided by the merchant using their debit and credit cards.

URLs:
https://api. test.oneroadpayments.com/merchants/{*merchantID*}/transactions
https://api.oneroadpayments.com/merchants/{*merchantID*}/transactions

Request Method:
POST

URI Parameters:
Required:

- *merchantID*
  uniquely identifies a merchant within the One Road Payments Gateway system.

Data Parameters:
Required:

- *clientID*[1]
  unique client identifier, provided by the merchant at time of registration. (must be unique, per merchant).

- *orderID*
  unique order identifier, supplied by the merchant with each payment. (must be unique, per merchant).

- *amount*
  sale amount for this payment transaction.

- *cardType*
  the card processing "type" that is to be used in this Payment. Currently the only supported values are "UnionPay" and "Wechat Pay".

- *currency*
  currency identifier, associated with transaction amount.

- *signature*
  cryptographic hash for this payment transaction.

- *signatureVersion*
  indicates the signature version used to ensure the integrity and authorisation for this transaction.

Optional:

- *echo*
  Value echoed back at the point of notification. This is not used during transaction authorization.

- *replyURL*
  When supplied, this optional parameter is used to override the default notification URL (registered in

---

[1] In the case of 'walk-up' or 'guest' clients, a merchant may use *Guest* as the client identifier. This will enable the interaction with unregistered participants. All other parameters are still required, as normal.

the merchant's settings).

HTML/Javascript Sample Code:

```
<!doctype html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge, chrome=1">
    <meta name="description" content="One Road Payments Gateway">
    <meta name="author" content="One Road Payments">

    <title>One Road Payments Gateway</title>
    <base href="./">
    <script src="https://cdn.jsdelivr.net/npm/crypto-js@3.1.9-1/crypto-js.min.js"></script>
    <script src="https://cdn.jsdelivr.net/npm/crypto-js@3.1.9-1/md5.js"></script>
    <script>
        function sign() {
            var key = '0246897531';
            var clearTxt = document.getElementsByName("clientID")[0].value +
                document.getElementsByName("orderID")[0].value +
                CryptoJS.MD5(key).toString() +
                document.getElementsByName("amount")[0].value +
                document.getElementsByName("currency")[0].value;
            if ( document.getElementsByName("replyURL")[0].value.trim() != "" )
                clearTxt = clearTxt + document.getElementsByName("replyURL")[0].value;
            document.getElementById("signature").value = CryptoJS.MD5(clearTxt).toString();
            return true;
        }
    </script>
</head>

<body>
    <form target="_blank" onsubmit="return sign()"
action="https://api.test.oneroadpayments.com /merchants/8873abde-7b40-4ac2-8774-
be331fa80ffd/transactions" method="POST">
        <input type="hidden" name="clientID" value="Customer 001">
        <input type="hidden" id="signature" name="signature" value="">
        <input type="hidden" name="signatureVersion" value="1.0">
        <div class="form-group">
            <label for="OrderID">Order Number</label>
            <input type="text" autofocus placeholder="Order Number" name="orderID" value="1">
        </div>
        <div class="form-group">
            <label for="replyURL">URL</label>
            <input type="text" placeholder="URL" name="replyURL" value="">
        </div>
        <div class="form-group">
            <label for="Amount">Amount</label>
            <input type="text" placeholder="Value" name="amount" value="1.00">
        </div>
        <div class="form-group">
            <label for="Currency">Currency</label>
            <select class="form-control" name="currency">
                <option>CYN</option>
                <option selected>USD</option>
                <option>EUR</option>
                <option>GBP</option>
                <option>JPY</option>
                <option>KRW</option>
                <option>AUD</option>
                <option>RUB</option>
                <option>CHF</option>
                <option>HKD</option>
                <option>SGD</option>
                <option>MOP</option>
                <option>CAD</option>
                <option>TWD</option>
            </select>
        </div>
        <div class="form-group">
            <label for="cardType">Card Type</label>
            <select class="form-control" name="cardType">
                <option selected>Union Pay</option>
                <option>Wechat Pay</option>
                <option>Visa</option>
```

```
                    <option>MasterCard</option>
                    <option>American Express</option>
                    <option>JCB</option>
                </select>
            </div>
        <button type="submit">Submit</button>
    </form>
</body>

</html>
```

Notes:

The sample code provided, above, shows a very simple HTML page used to initiate a payment, via One Road Payments Gateway.

The important elements here are:

1) In the <header>, Lines 12 and 13, importing the crypto libraries for performing MD5 hashing.

2) The supplied script from lines 14 to 27 show how to calculate the required signature for a payment request. Within this function there is a variable, key. The key value shown is that used for testing and integration. Each merchant will have a statistically unique key, agreed with One Road Payments, when moving to production. The clear text is merely a concatenation of: client identifier, order number, a salt (derived from the clear key), amount, currency and optionally, the replyURL.

3) The replyURL is an optional parameter that overrides the default notification URL, associated with the merchant record.

4) The body shows the URL used for payment submission, with the unique merchant ID embedded within it.

5) The supplied HTML source will submit the client identifier, merchant payment signature, signature version, order number, transaction amount and currency, all of which are mandatory.

6) Upon form submission, via HTTP POST, the client's browser will be redirected to a new webpage, through which they will provide the necessary credentials for Issuer authentication and account authorisation.

PHP Sample Code:

Payment.php

```
<?php

class payment
{
    static $security_key    = "0246897531"; //merchant encryption key
    static $signVersion     = "1.0";

    //* Test environment
    static $pay_url         = "http://api.test.oneroadpayments.com/merchants/8873abde-7b40-
4ac2-8774-be331fa80ffd/transactions";

    static function sign($params) {
            $sign_str = get($params, "clientID");
            $sign_str .= get($params, "orderID");
            $sign_str .= md5($security_key);
            $sign_str .= get($params, "amount");
            $sign_str .= get($params, "currency");
            return md5($sign_str);
    }

    static function get($params, $key) {
        if (isset($params[$key])) {
```

```php
            return $params[$key];
        } else {
            throw new Exception("missing parameter: " . $key);
        }
    }
}

?>
```

Payment_service.php

```php
<?php

//Payment interface example

require_once('./payment_service.php');

$params['clientID']             = "Customer 001";           // Idenifier for the registered
client.
$params['amount']               = "1.00";                   // This is the trasnaction Amount
=> $HKD1.00
$params['orderID']              = "1";                      // This needs to be unique, for
each transaction.
$params['currency']             = "HKD";                    // Currency of the transaction
$params['cardType']             = "UnionPay";               // Card scheme
$params['signatureVersion']     = payment::$signVersion;    // Version of the signature
algorithm

$pay service = new payment();
$params['signature'] = $pay service($params);

$html = <<<eot
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <script
language="javascript">window.onload=function(){document.pay form.submit();}</script>
</head>
<body>
    <form id="pay form" name="pay form" action="{$pay service->pay url}" method="POST">
eot;
        foreach ($params as $key => $value) {
            $html .= "  <input type=\"hidden\" name=\"{$key}\" id=\"{$key}\"
value=\"{$value}\"/>\n";
        }
        $html .= <<<eot
    </form>
</body>
</html>
eot;

header("Content-Type: text/html; charset=UTF-8");
echo $html;

?>
```

Notes

It should be noted that the client's browser will be redirected back to the URL stored in the merchant settings, within One Road Payments Gateway. At the point of notification, the return URL will have the following post fixed to it:

?success=Y&orderID=1&clientID=Customer+001&amount=1.00&currency=HKD&signatu re=a599afb474693f6b9efea5579ff769a6

The signature attached here is to validate that the response is being transmitted from One Road Payments. It is created in a very similar way to the signature submitted with the payment request document. The only differences here are that the clear text is prefixed with

success value (in the above example, a 'Y') , and the merchant ID is appended to the clear text.

Query

(If the supporting information for this call is not supplied below, users should contact One Road Payments)

This function is used to validate the status of a previously executed Purchase. Typically called when a client queries the final outcome of their payment.

URLs:
https://api.test.oneroadpayments.com/merchants/{*merchantID*}/transactions/{orderID}
https://api.oneroadpayments.com/merchants/{*merchantID*}/transactions/{orderID}

Request Method:
GET

URI Parameters:
Required:
- *merchantID*   uniquely identifies a merchant within the One Road Payments Gateway system.
- *orderID*   uniquely identifies a payment within the One Road Payments Gateway system (previously supplied in the original Purchase).

Data Parameters:
Required:
- *None*

HTML/Javascript Sample Code:

```
<!doctype html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge, chrome=1">
    <meta name="description" content="One Road Payments Gateway">
    <meta name="author" content="One Road Payments">

    <title>One Road Payments Gateway</title>
    <base href="./">
    <script src="https://cdn.jsdelivr.net/npm/crypto-js@3.1.9-1/crypto-js.min.js"></script>
    <script src="https://cdn.jsdelivr.net/npm/crypto-js@3.1.9-1/md5.js"></script>
    <script>
        function sign() {
            var key = '0246897531';
            var clearTxt = document.getElementsByName("clientID")[0].value +
                document.getElementsByName("orderID")[0].value +
                CryptoJS.MD5(key).toString() +
                document.getElementsByName("amount")[0].value +
                document.getElementsByName("currency")[0].value;
            if ( document.getElementsByName("replyURL")[0].value.trim() != "" )
                clearTxt = clearTxt + document.getElementsByName("replyURL")[0].value;
            document.getElementById("signature").value = CryptoJS.MD5(clearTxt).toString();
            return true;
        }

        function query() {
            var request = new XMLHttpRequest();
            request.open('GET', 'https://api.test.oneroadpayments.com/merchants/8873abde-7b40-
4ac2-8774-be331fa80ffd/transactions/' +
document.getElementsByName("OrderID")[0].value.toLowerCase(), true);
            request.setRequestHeader("x-api-key", "i74wnDTxoi4Nxy61aatbr6mNj2KQtDb31iqrQJ1c");
            request.onload = function() {
                var data = JSON.parse(this.response);
                if (request.status != 200) {
                    document.getElementById("message").innerHTML = data.response.detail;
                } else {
                    var message = 'Success: ' + data.reply.success;
                    if (data.reply.success == 'Y') {
                        message += '<br><br>URL: ' + data.reply.url;
                    }
                    document.getElementById("message").innerHTML = message;
                }
            }
            request.send();
        }
    </script>
</head>

<body>
    <form target=" blank" onsubmit="return sign()"
action="https://api.test.oneroadpayments.com /merchants/8873abde-7b40-4ac2-8774-
be331fa80ffd/transactions" method="POST">
        <input type="hidden" name="clientID" value="Customer 001">
        <input type="hidden" id="signature" name="signature" value="">
        <input type="hidden" name="signatureVersion" value="1.0">
        <div class="form-group">
            <label for="OrderID">Order Number</label>
            <input type="text" autofocus placeholder="Order Number" name="orderID" value="1">
        </div>
        <div class="form-group">
            <label for="replyURL">URL</label>
            <input type="text" placeholder="URL" name="replyURL" value="">
        </div>
        <div class="form-group">
            <label for="Amount">Amount</label>
            <input type="text" placeholder="Value" name="amount" value="1.00">
        </div>
        <div class="form-group">
```

```
            <label for="Currency">Currency</label>
            <select class="form-control" name="currency">
                            <option>CYN</option>
                            <option selected>USD</option>
                            <option>EUR</option>
                            <option>GBP</option>
                            <option>JPY</option>
                            <option>KRW</option>
                            <option>AUD</option>
                            <option>RUB</option>
                            <option>CHF</option>
                            <option>HKD</option>
                            <option>SGD</option>
                            <option>MOP</option>
                            <option>CAD</option>
                            <option>TWD</option>
                        </select>
        </div>
        <div class="form-group">
            <label for="cardType">Card Type</label>
            <select class="form-control" name="cardType">
                <option selected>Union Pay</option>
                <option>Wechat Pay</option>
                <option>Visa</option>
                <option>MasterCard</option>
                <option>American Express</option>
                <option>JCB</option>
            </select>
        </div>
        <p class="form-control" id="message"></p>
        <button type="submit">Submit</button>
        <button type="button" onclick="query()">Query</button>
    </form>
</body>

</html>
```

Notes:

The sample code provided, above, shows a very simple HTML page used to initiate a payment and corresponding query, via One Road Payments Gateway.

The important elements here are:

1) In the <header>, Lines 12 and 13, importing the crypto libraries for performing MD5 hashing.

2) The supplied script from lines 14 to 27 show how to calculate the required signature for a payment request. Within this function there is a variable, key. The key value shown is that used for testing and integration. Each merchant will have a statistically unique key, agreed with One Road Payments, when moving to production. The clear text is merely a concatenation of: client identifier, order number, a salt (derived from the clear key), amount, currency and optionally, the replyURL.

3) The reply URL is an optional parameter that overrides the default notification URL, associated with the merchant record.

4) The body shows the URL used for payment submission, with the unique merchant ID embedded within it.

5) The supplied HTML source will submit the client identifier, merchant payment signature, signature version, order number, transaction amount and currency, all of which are mandatory.

6) Upon form submission, via HTTP POST, the client's browser will be redirected to a new webpage, through which they will provide the necessary credentials for Issuer authentication and account authorisation.

It should be noted that the client's browser will be redirected back to the URL stored in the merchant settings, within One Road Payments Gateway. At the point of notification, the return URL will have the following post fixed to it:

?success=Y&orderID=1&clientID=Customer+001&amount=1.00&currency=HKD&signature=a599afb474693f6b9efea5579ff769a6

The signature attached here is to validate that the response is being transmitted from One Road Payments. It is created in a very similar way to the signature submitted with the payment request document. The only differences here are that the clear text is prefixed with *success* value (in the above example, a 'Y'), and the merchant ID is appended to the clear text.

Void/Purchase Cancellation

(If the supporting information for this call is not supplied below, users should contact One Road Payments)

This call is used to cancel a purchase made prior to the to the merchant being settled for the purchase.

URL:

Not Supplied

Request Method:

Not Supplied

URI Parameters:

Required:

Not Supplied

Optional:

Not Supplied

Data Parameters:

Required:

Not Supplied

Optional:

Not Supplied

Success Response:

Not Supplied

Error Response:

Not Supplied

Sample Code:

Not Supplied

Notes:

Refund

(If the supporting information for this call is not supplied below, users should contact One Road Payments)

This call is used to refund money to a client's card after the merchant has been settled for the purchase.

URL:

Not Supplied

Request Method:

Not Supplied

URI Parameters:

Required:

Not Supplied

Optional:

Not Supplied

Data Parameters:

Required:

Not Supplied

Optional:

Not Supplied

Success Response:

Not Supplied

Error Response:

Not Supplied

Sample Code:

Not Supplied

Notes:

5. GLOSSARY AND DEFINITIONS

Acquirer or Payment Processor. The organisation which holds the merchant relationship and facilitates payments on behalf of the merchant.

Authorization. A request transaction captured from a merchant, merchant processor or Point-of-Sale device that requires routing to an authorizing institution for the response.

Batch. Refers to a group of merchant transactions. One Road Payments Gateway stops each day's processing batch at a set time, opening a new batch for the next day's transactions. Note that cutover time of the batch may not be in line with merchant operating hours.

Cardholder. The customer to whom a card has been issued, the customer of the merchant.

Credit/Debit Authorization. A transaction that fulfils a request to an issuer to validate the availability of funds for payment against a line of credit or account that is accessed by a payment card (credit/debit card).

Credit/Debit Authorization Request. The request portion of a Credit/Debit Authorization from the One Road Payments Gateway system to an issuer.

Credit/Debit Authorization Response. The response portion of a Credit/Debit Authorization from an issuer institution to the One Road Payments Gateway system.

Change Period. This is the period of time when scheduled changes to the One Road Payments Gateway systems and applications are to occur.

Cutover. The action of changing from one business day to the next. Cutover is the process of rolling the business day of the One Road Payments Gateway systems to the next day. The cutover time can be configured to any time during a 24-hour day.

Endpoint. The destination to which a financial interchange transaction is delivered, which may be connected to the One Road Payments Gateway system. A telecommunication address that the One Road Payments Gateway switch uses to route a transaction.

Establishment. Establishment is the definition, development and implementation of the specified service delivery capability. It does not imply the delivery of services to the merchants. Establishment is required prior to merchant enablement.

Issuer. The institution which issues the card to the cardholder and approves the transaction from One Road Payments Gateway.

Merchant Administration System. A secure, internet portal which allows merchants to manage their online processing.

Merchant Agreement. The contractual agreement between acquirers or processors and participating merchants for the purpose of facilitating merchant payment acceptance.

Merchant Interface Software. Software integrated into the merchant's payment system to enable connectivity to the One Road Payments Gateway system.

Refund. A transfer of funds from the merchant to the cardholder. Refunds must be matched to a purchase transaction and must not exceed the value of the original transaction.

Routing. The act of determining the destination of a financial interchange transaction request. The One Road Payments Gateway system routes a transaction by determining what party will receive it, and then acts upon that information to deliver the request to the determined endpoint.

Settlement. The completion of a transaction or set of transactions with the aim of discharging participants' obligations through the transfer of funds. Settlement can involve the transfer of funds from an issuer to an acquirer or processor or from an acquirer or processor to a merchant.

SSL. Secure Socket Layer is a data security standard which encrypts data between a web browser and a web server.

Switch. A term that refers to the One Road Payments Gateway system.

Switch Cutover. The time designated to change from one business day to the next for the One Road Payments Gateway system (see Cutover).

One Road Payments Gateway Merchant Administration Service. An online accessible portal for merchants to be able to manage their transactions and download reports and relevant data.

Third Party Processor or Payment Service Provider (PSP). Non-bank or third-party service provider of data switching services on behalf of acquirers or issuers within the financial interchange arena.

User Accounts. Accounts provided to allow secure access to the *One Road Payments Gateway Merchant Administration Service* through the use of a user ID and password. These are Merchant User Accounts, available for use by merchants.

One Road Payments Gateway Product Service Guide. Documentation describing the One Road Payments Gateway standard message formats and data flows for *Credit/Debit Authorization* transaction interchange (this document).