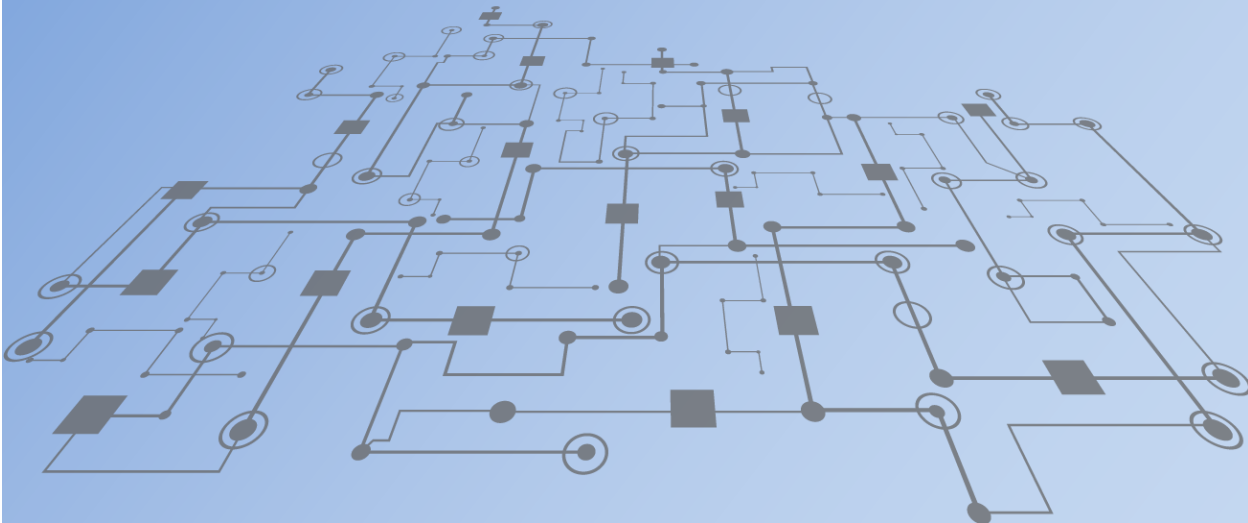


# Open API Banking Utility Pay

v1.0

All Rights Reserved ©2019, Nations Trust Bank PLC



---

## Table of Contents

1.0	Introduction .....	2
1.1	Purpose .....	2
1.2	Intended audience and reading suggestions .....	2
2.0	General Information .....	2
2.1	Authentication and authorization .....	2
2.1.1	Retrieve Access Token .....	2
2.1.2	The Cryptography Mechanism .....	3
3.0	API Flow .....	5
4.0	Recourses .....	6
4.1	Utility Bill Payment Pre-Process .....	6
4.1.1	Sample Request .....	7
4.1.2	Sample Response .....	7
4.2	Utility Bill Payment Confirm-Process .....	8
4.2.1	Sample Request .....	8
4.2.2	Sample Response .....	9

## 1.0 Introduction

### 1.1 Purpose

The purpose of the document is to describe Open API Banking resources and how to correctly access them.

### 1.2 Intended audience and reading suggestions

The intended audience for this document comprises mainly software developers, software development managers or any other person with technical expertise in software development.

## 2.0 General Information

### 2.1 Authentication and authorization

Security model of Open API Banking is based on SSL usage and presence of Authorization header in each request containing a security token. Token will have 5 minute expiry time in live. But in test system, expiry is 1 hour.

#### 2.1.1 Retrieve Access Token

**POST** <https://uatopenapi.nationstrust.com:8243/token/>

\*\*This URL is for development environment only. The bank will issue a separate URL for the production.

#### Authorization

Type **Basic Auth**

**Username** string **Required**

HTTP Basic authentication by passing encoded value of the user name.

*Ex: ZuLJkn3DqzEjQshfnkMaN3VqjjEa*

*\*\*Will be provided after the access account creation.*

---

#### Password

**string** Required

HTTP Basic authentication by passing encoded value of the password.

*Ex: OveDqjfNX8gn1H2Fg1\_dQ4ZjmJQa*

*\*\*Will be provided after the access account creation.*

---

#### Header Parameters

##### Content-Type

**string** Required

Content-Type that are acceptable for the response.

*Ex: application/x-www-form-urlencoded*

---

#### Form Data Parameters

##### grant\_type

**string** Required

The grant type for each client.

*Ex: client\_credentials*

---

Production Username and Password will be providing for each merchant when it's ready to go live. For testing, above test data can be used.

### 2.1.2 The Cryptography Mechanism

The system has implemented with one of the latest cryptography mechanism AES/RSA encryption. The RSA key size is 1024.

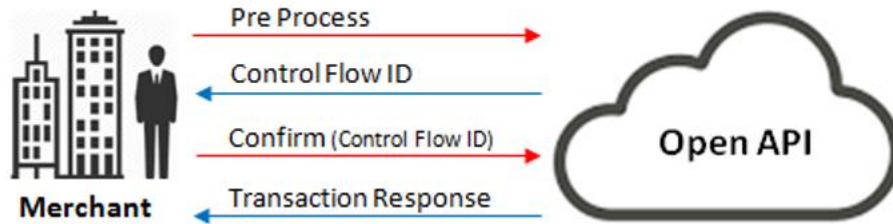
Each client who uses the API will receive his own unique System ID (*mid*) and *RSA* public key. This key needs to encrypt sensitive data like account numbers, transaction amount & other required fields. The clients should be careful to not to share those on any digital media due to safety matter.

Please refer the given code snip below, to understand the encryption mechanism used in **C#** environemt.

```
public string Encryption(string data, string publicKey)
{
    using (var rsa = new RSACryptoServiceProvider(1024))
    {
        try
        {
            rsa.FromXmlString(publicKey.ToString());

            var dataToEncrypt = Encoding.UTF8.GetBytes(data);
            var encryptedByteArray = rsa.Encrypt(dataToEncrypt, false).ToArray();
            return Convert.ToBase64String(encryptedByteArray);
        }
        finally
        {
            rsa.PersistKeyInCsp = false;
        }
    }
}
```

### 3.0 API Flow



**\*\*Important:** Please make sure that, all the API requests related to “fund transfers & utility payments” should have mandatory fields in the http header as mentioned below,

#### Header Parameters

**mid**

**string** Required

A unique key released by the bank to identify merchants.

*Ex: 100001*

**uuid**

**string** Required

10 digit number used for identifying transaction uniquely. Should have a unique id for a particular day.

*Ex: 10010015*

**Content-Type**

**string** Required

The data structure to get the API response.

*Ex: application/json*

**Authorization**

**string** Required

Need to mention the token\_type and the access token separated by a space. `{ token_type } { access_token }`

*Ex: Bearer e0386644-0820-3f4d-a8a0-cca379324b07*

## 4.0 Recourses

### 4.1 Utility Bill Payment Pre-Process

POST

<https://uatopenapi.nationstrust.com:8243/openbanking/ubpayment/preprocess/v1>

\*\*This URL is for development environment only. The bank will issue a separate URL for the production.

#### Form Data Parameters

sourceAccountId

string Required Encryption Required

The account number where the debit process should happen on. Client should encrypt the account number using the key provided by the bank.

Max-length : 7 *(before the encryption)*

Ex: XXXXXXX

txnAmount

string Required Encryption Required

The amount should pass to the transaction process.

Ex: 250.50

billerId

string Required Encryption Required

The unique ID to identify each drilling category. The bank will provide a list of billers and relevant information.

Max-length : 5 *(before the encryption)*

Ex: 435

referenceNo

string Required

This parameter should be a unique one and it is really important to track the transaction happened in the baking system.

Max-length : 25 *(before the encryption)*

Ex: TXN234000234523

#### remarks

#### string

This parameter collects the purpose of the credit transaction. It's better to have a meaningful content for review.

Max-length : 30 (*before the encryption*)

*Ex: tele bill payment*

#### 4.1.1 Sample Request

The body section of the POST request should contain the JSON request according to following JSON sample. Please make sure to use the encrypted data on relevant fields explained in the above section.

```
{
  "sourceAccountId": "XXXXXXXX",
  "txnAmount": "100",
  "billerId": "435",
  "referenceNo": "TXN234443000034465",
  "remarks": "reload"
}
```

*Note: Sensitive data should be encrypted using AES / RSA mechanism according to above section.*

#### 4.1.2 Sample Response

In the response, you can collect the "controlFlowId" which is needed to perform the transaction confirm call in the next step.

```
{
  "controlFlowId":
  "8DB66A38E84547DCE05400144FF8DFBF",
  "debitDetails": {
```



```
        "txnDebitAmount": "10",
        "currencyCode": "LKR"
    },
    "creditDetails": {
        "txnCreditAmount": "10",
        "currencyCode": "LKR"
    },
    "billerName": "Dialog Broadband",
    "referenceNo": "0712112326"
}
```

## 4.2 Utility Bill Payment Confirm-Process

POST

<https://uatopenapi.nationstrust.com:8243/openbanking/ubpayment/confirm/v1>

\*\*This URL is for development environment only. The bank will issue a separate URL for the production.

### Form Data Parameters

**controlFlowId**

string Required

“controlFlowId” id return from pre-process request

*Ex: 8DB66A38E84547DCE05400144FF8DFBF*

### 4.2.1 Sample Request

The body section of the POST request should contain the JSON request according to following JSON sample. In this call “controlFlowId” not required to encrypt.

```
{
  "controlFlowId": "8DB66A38E84547DCE05400144FF8DFBF "
}
```

#### 4.2.2 Sample Response

For successful transactions, the service will return the following JSON response.

```
{
  "Message": "Sucess",
  "ModelState": "{}",
  "ResCode": "00",
  "UUID": "111111106012",
  "ReferenceNo": "Successful-018727"
}
```