



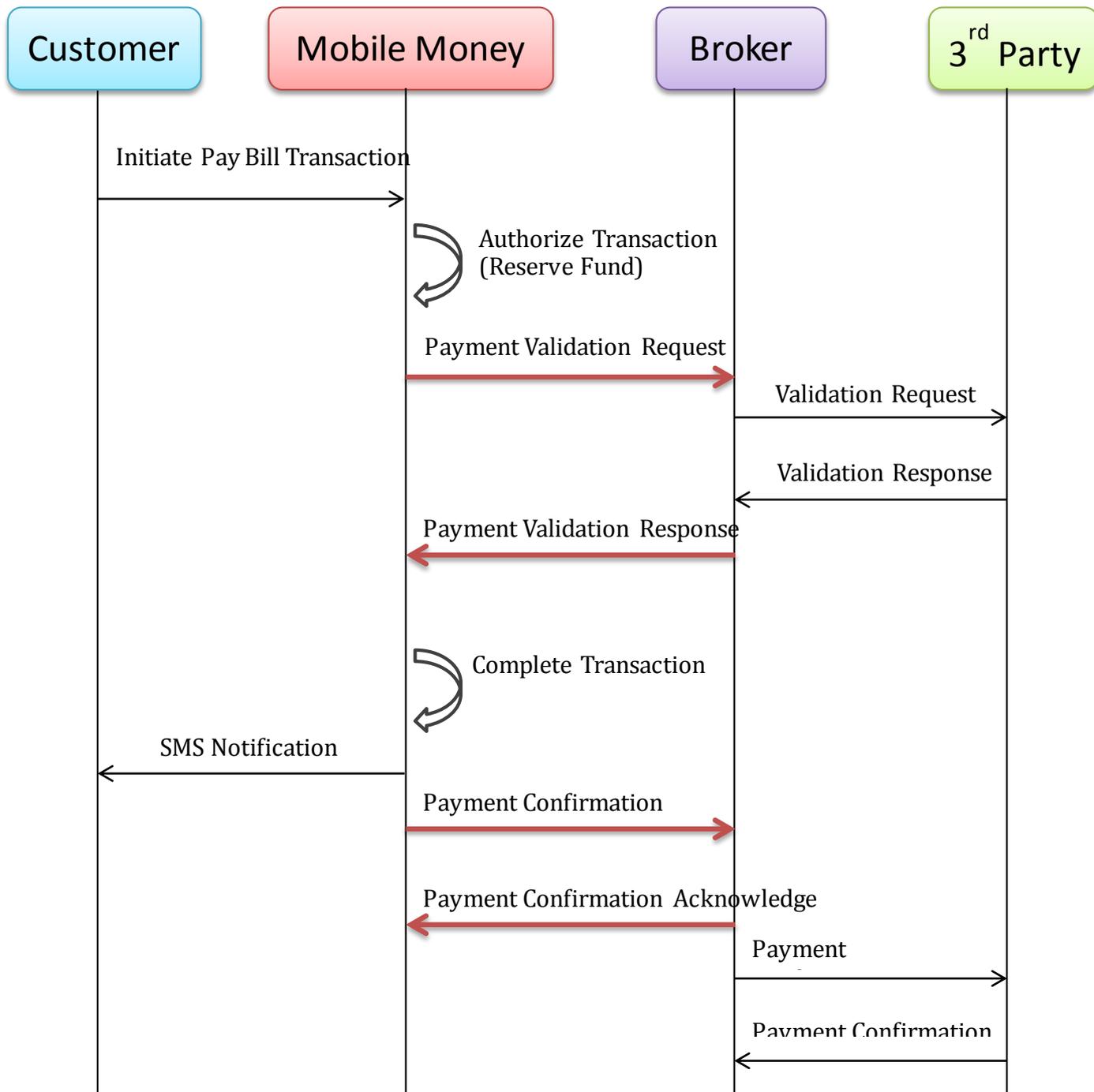
Safaricom Integration Guide (C2B) – Quick Setup

Table of Contents

Introduction.....	2
Step 1: What you will require for testing	3
Step 2: Setting up the listener.....	4
Step 3: Testing.....	4
3.1 HTTP TEST.....	4
3.2 MOVING TO HTTPS (2-Way SSL)	4
Step 4: The setup.....	6
4.1: Register URL.....	6
Step 4.2: Confirmation and Validation	7
Confirmation.....	7
Validation.....	8
Step 5: Security (SOAP request Header)	9
Step 6: Data Type Definitions.....	9
Step 7: The SOAP Request Body	10
Error Codes.....	11
Command id's	12
Finalizing the integration	12
FAQ's	12

Introduction

This document describes the steps to be followed when testing on M-Pesa for a customer to business (Paybill and buygoods) transaction. The following diagram describes the process flow:



Step 1: What you will require for testing

This is an important step as it allows us to gather the requirements for proceeding with the tests. The following will be used for testing and should be installed:

Firstly, you will need to know how you are going to connect to us.

	IT VPN	EXTRANET VPN	DATA (P2P)	ports
Test bed	196.201.214.136	10.26.0.75	192.168.9.48	8310 - http, 18423 – https.
Production Request	196.201.214.137	10.26.0.76	192.168.9.49	8310 - http, 18423 – https.
Production Result	196.201.214.127	10.26.0.77	192.168.9.50	8310 - http, 18423 – https.

For broker end points for testing using the IP and the ports above the general format will be:

http: <http://IPAddress:port/mminterface/request>

https: <https://IPAddress:port/mminterface/request>

http: <http://IPAddress:port/mminterface/registerURL>

https: <https://IPAddress:port/mminterface/registerURL>

http: <http://IPAddress:port/queryTransactionService/services/transaction>

https: <https://IPAddress:port/queryTransactionService/services/transaction>

- You will be sent a folder called Third Party v3. This will include the WSDL files and documents you will need to use together with SOAP. You can use the link below to download them: https://drive.google.com/folderview?id=0B_BRYv-DOjeTamo3TFVBX2VpcTQ&usp=sharing.
- You will need to have SOAP installed to send request and see responses from broker.
- Ensure you have shared your source IP with us.
- Ensuring you have a Test Paybill, service id, username and password assigned to carry out test transactions, otherwise please contact the FS Planning team on FSPlanning@safaricom.co.ke and it shall be created. However, you will need to send your source IP and the SPid you are going to use to the email above.
- You can use the default testing credentials
- Editing and requests will be sent and done on one particular WSDL called CBP_Interface_Request>GenericAPI>C2B.

Step 2: Setting up the listener

The listener is used to process events as they occur. Thus, for testing purposes to receive responses for successful and erroneous transactions, we will be required to set up a listener:

- For setting up the listener, refer to step 4 below to set up confirmation and validation.
- You then register the URL's that you set up (in step 4) as shown in Step 3 (Register URL) below.

Step 3: Testing

After importing all the WSDL's to SOAP, you shall begin with setting up the endpoints. Depending on your connection to us, refer to step 1. Ensure you can telnet to the endpoints that you are connected to confirm connectivity.

3.1 HTTP TEST

Ensure your server configurations and ports are on http. Please inform us via mail on the ports that you will be going to use.

1. Perform a telnet and share output for both production and test bed IP and ports.
2. Send request and check for any errors.
3. Confirm that you are able to post a successful request and check for a successful response on your callback URL.
4. Confirm that you have configured Callback URL (Result URL) in your request to receive GenericAPIResult from broker with the IP and ports that are configured in the firewall on our side.

3.2 MOVING TO HTTPS (2-Way SSL)

Generate the Certificate Signing Request (CSR) depending on the web server platform while strictly adhering to the following guidelines:

The following fields must be populated with valid and verifiable information:

- i. Common Name= **FQDN of hosting server using standard notation**
- ii. Organization Name= **Business Name as recognized by Safaricom**
- iii. Locality= **Region where the organization is located**
- iv. State / Province= **Province/County**
- v. Country = **Country of registration of the organization**
- vi. Minimum Key Size is = **2048bits**

2. Important Considerations for Certificate Requestors:

- i. The common name needs to be a Fully Qualified Domain Name (FQDN) and unique for every request made. Even if the name is not verifiable via public DNS, please select a similar name anyway

that follows standard DNS notation since other arrangements can be made for the mapping of that name to the corresponding IP address e.g. **endpoint1.3rdpartyname.co.ke**

ii. The private key generated along with the CSR is solely for the Requestor to secure. We do encourage that you protect this key with a password and ensure relevant access controls are put in place for the server that will be hosting the private key. This key should NOT be shared with anyone. Any compromise of the key shall be communicated immediately to Safaricom and a new one generated so that a new certificate can be generated.

iii. Safaricom will issue certificates with a maximum validity of 2 years. The requestor will be responsible for tracking the expiry of the certificate issued and raise a new request at least 2 months before the expiry of the certificate.

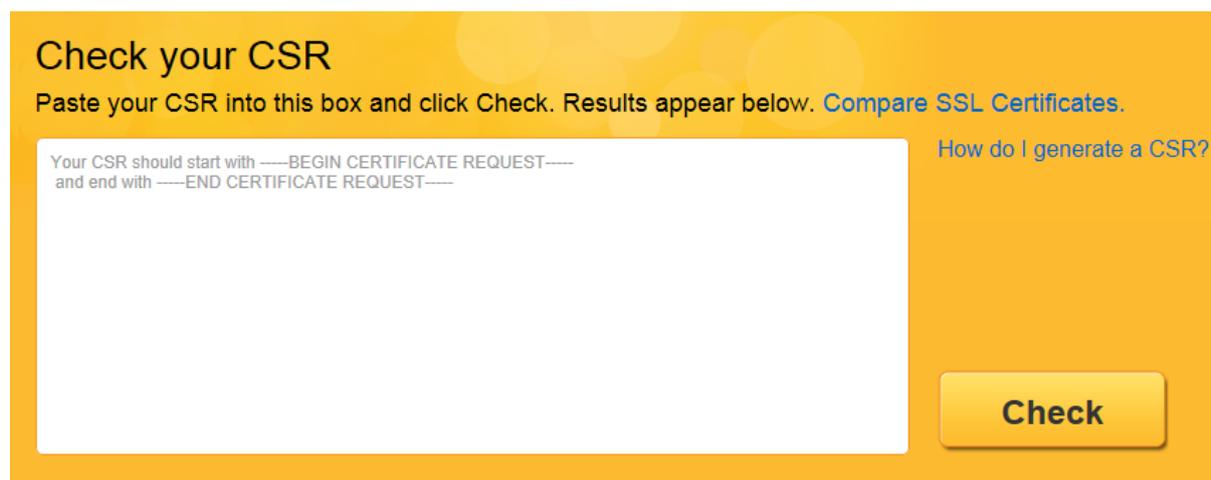
iv. Certificates generated are solely for the private use when connecting to Safaricom systems and for the purpose intended as prescribed by Safaricom. Breach of acceptable usage for the certificate issued will result in immediate revocation and possible legal action if deemed necessary by Safaricom.

3. Checking for Errors on the CSR:

i. Open Symantec CSR checker to make sure the CSR has been generated properly using the link <https://cryptoreport.websecurity.symantec.com/checker/views/csrCheck.jsp>

ii. Copy and paste the contents of the CSR in the box then click check to see if there are any errors.

iii. Submit your CSR for signing after checking and correcting errors



Check your CSR

Paste your CSR into this box and click Check. Results appear below. [Compare SSL Certificates.](#)

Your CSR should start with ---BEGIN CERTIFICATE REQUEST---
and end with ---END CERTIFICATE REQUEST---

[How do I generate a CSR?](#)

Check

Steps:

- Change your ports to https and also inform us of the change on our end.

- Once the above is done you can send the .csr to FSPlanning@safaricom.co.ke for signing.

- Once you have the signed certificate, configure testbroker.crt added as a CA in the local trust store, this will be used to authenticate broker.


```

        <ThirdPartyID/>
        <Password/>
        <Checksum/>
        <ResultURL/>
    </Caller>
    <Initiator>
        <IdentifierType>l</IdentifierType> //Constant variable,remains the same
        <Identifier/>
        <SecurityCredential/>
        <ShortCode/>
    </Initiator>
    <PrimaryParty>
        <IdentifierType>l</IdentifierType> //Constant variable, remains the same
        <Identifier/>
        <ShortCode>0000111</ShortCode> //your short code
    </PrimaryParty>
</Identity>
    <KeyOwner>l</KeyOwner> //Constant variable, remains the same.
</request>]]></req:RequestMsg>
</soapenv:Body>
</soapenv:Envelope>

```

✓ Response

The following is a sample of the response that is expected after the request for registering URL.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:req="http://api-v1.gen.mm.vodafone.com/mminterface/request">
    <soapenv:Header/>
    <soapenv:Body>
        <req:ResponseMsg><![CDATA[<?xml version="1.0" encoding="UTF-8"?><response
xmlns="http://api-
v1.gen.mm.vodafone.com/mminterface/response"><ResponseCode>00000000</ResponseCode><ResponseDes
c>success</ResponseDesc><ConversationID></ConversationID><OriginatorConversationID></Originato
rConversationID><ServiceStatus>0</ServiceStatus></response>]]></req:ResponseMsg> //response 0
is success, anything other than 0 is an error.
    </soapenv:Body>
</soapenv:Envelope>

```

Step 4.2: Confirmation and Validation

For some companies, confirmation and validation plays a critical part in their systems. You will need to import the following:

-The WSDL CBPIInterface_C2BPaymentValidationandConfirmation

Confirmation

The following is the general structure used for confirmation for a C2B transaction:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:c2b="http://cps.huawei.com/cpsinterface/c2bpayment">
    <soapenv:Header/>
    <soapenv:Body>
        <ns1:C2BPaymentConfirmationRequest>
            <TransactionType>PayBill</TransactionType> //Transaction type PayBill
            <TransID>1234560000007031</TransID> //Transaction ID
            <TransTime>20140227082020</TransTime> //Transaction time
            <TransAmount>123.00</TransAmount> //Transaction amount
            <BusinessShortCode>12345</BusinessShortCode> //Business Short code
            <BillRefNumber>TX1001</BillRefNumber> //BillRefNumber (on user interface Account No.)
            <InvoiceNumber></InvoiceNumber> //Invoice number, if required
            <OrgAccountBalance>12345.00</OrgAccountBalance> //Organization account balance
            <ThirdPartyTransID></ThirdPartyTransID> //Third party transID
        </ns1:C2BPaymentConfirmationRequest>
    </soapenv:Body>
</soapenv:Envelope>

```

```

<MSISDN>254722703614</MSISDN> //MSISDN sending the request
  <KYCInfo>
    <KYCName>[Personal Details][First Name]</KYCName> //KYC info (first
name)
    <KYCValue>Hoiyor</KYCValue>
  </KYCInfo>
  <KYCInfo>
    <KYCName>[Personal Details][Middle Name]</KYCName> //KYC info (Middle
name)
    <KYCValue>G</KYCValue>
  </KYCInfo>
  <KYCInfo>
    <KYCName>[Personal Details][Last Name]</KYCName> //KYC info (Last name)
    <KYCValue>Chen</KYCValue>
  </KYCInfo>
</ns1:C2BPaymentConfirmationRequest>
</soapenv:Body>
</soapenv:Envelope>

```

Below is a sample response that you shall get after a successful confirmation has been carried out

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:c2b="http://cps.huawei.com/cpsinterface/c2bpayment">
  <soapenv:Header/>
  <soapenv:Body>
    <ns1:C2BPaymentConfirmationResult>C2B Payment Transaction 123456000007031 result
received.</ns1:C2BPaymentConfirmationResult> //Result received indicates success, otherwise
failed.
  </soapenv:Body>
</soapenv:Envelope>

```

Validation

The general format used for the validation of a C2B request:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:c2b="http://cps.huawei.com/cpsinterface/c2bpayment">
  <soapenv:Header/>
  <soapenv:Body>
    <ns1:C2BPaymentValidationRequest>
      <TransactionType>PayBill</TransactionType> //Transaction type is paybill
      <TransID>123456000007031</TransID> //Transaction ID
      <TransTime>20140227082020</TransTime> //Transaction Time
      <TransAmount>123.00</TransAmount> //Transaction Amount
      <BusinessShortCode>12345</BusinessShortCode> //The short code for the business
      <BillRefNumber></BillRefNumber> //BillRefNumber (on user interface account number.)
      <InvoiceNumber></InvoiceNumber> //Invoice number, if required
      <MSISDN>254722703614</MSISDN> //MSISDN number for the request
      <KYCInfo>
        <KYCName>[Personal Details][First Name]</KYCName> //KYC info(first
name)
        <KYCValue>Hoiyor</KYCValue>
      </KYCInfo>
      <KYCInfo>
        <KYCName>[Personal Details][Middle Name]</KYCName> //KYC info (middle
name)
        <KYCValue>G</KYCValue>
      </KYCInfo>
      <KYCInfo>
        <KYCName>[Personal Details][Last Name]</KYCName> //KYC info (Last name)
        <KYCValue>Chen</KYCValue>
      </KYCInfo>
    </ns1:C2BPaymentValidationRequest>
  </soapenv:Body>
</soapenv:Envelope>

```

The following is a successful validation C2B Response:

2	TransID	String	123456000000 7031	No	The unique identifier of the payment transaction that is generated by M-Pesa system.
3	TransTime	String	YYYYMMDDhh mmss 201402270820 20	No	The date and time when customer initiates a Buy Goods transaction.
4	TransAmount	String	123.00	No	The amount of the transaction. Two decimals fixed point number. Fixed Currency Type: KES e.g. 123.00 stands for: 123.00 Ksh
5	BusinessShortCode	String	12345	No	The organization short code of the Merchant in the M-Pesa system
6	BillRefNumber	String	TX1001	Yes	This field is not applicable for Buy Goods transactions. It will always be blank.
7	InvoiceNumber	String	X334343	Yes	Invoice Number as Entered by Identity This field is reserved. It will always be blank in current delivery.
8	MSISDN	String	254722703614	No	The customer's MSISDN, which is with country code prefix.
	KYCInfo				The customer's KYC information, maybe multiple KYC info.
9	KYCName	String	[Personal Details][First Name]	Yes	This KYC indicates the customer's first name
10	KYCValue	String	Hoiyor	Yes	The value of KYC field
11	KYCName	String	[Personal Details][Middle Name]	Yes	This KYC indicates the customer's middle name
12	KYCValue	String	G	Yes	The value of KYC field
13	KYCName	String	[Personal Details][Last Name]	Yes	This KYC indicates the customer's last name
14	KYCValue	String	Chen	Yes	The value of KYC field

Step 7: The SOAP Request Body

This is how the final request should look like:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:req="http://api-
v1.gen.mm.vodafone.com/mminterface/request">
  <soapenv:Header>
```


Command id's

The following is a list of command id's that can be used with this integration.

BusinessBuyGoods
BusinessPayBill
CustomerBuyGoodsOnline

Finalizing the integration

1. Once the above is carried out on http (refer to step 3.1)
2. Carry the above out on https (refer to step 3.2), send signed off UAT's
3. Once testing is completed, the person assisting you will help you move to production.

FAQ's

1. Who initiates the transaction on C2B?
We initiate the transactions and check if you check for confirmation and validation on your side.
2. Do we require having a https connection?
Yes, as a compulsory requirement you do require a https after testing on http and before moving to the live platform
3. What is different on the live environment than the testing?
Everything is replicated on both platforms, the only thing different is you will be creating your own credentials and will have access to an M-Pesa web interface. You will not be provided the API credentials by us.
4. What is the easiest way to ask for support?
You can get in touch on google hangouts or skype with any of the members of the FS Planning team members that are assisting you.
5. Is there anything that can be changed on the API?
Unfortunately, since a lot of businesses are connected to the API, changes will be difficult as it will affect all the users.
6. Is there a way I can download all the files that I will require in the tests?
Yes, all the files that you will require apart from the .p7b's will be available at the following link: https://drive.google.com/folderview?id=0B_BRyv-DOjeTamo3TFVBX2VpcTQ&usp=sharing
7. Does the format of the request and response need to be the same as the .wsdl you sent us?
Yes, the format should remain the same. Any change will result in an error result.
8. When should we start testing?

You should start testing a month or more in advance with our teams so that any errors encountered can be quickly dealt with.