apigateMint API Specification [Northbound] - Identity:Link

Document Reference No.: APIGATE/MINT/APIS-NB-IDN-LINK

Version: 1.1

Issued: 8 March 2019

Table of Contents

Glossary

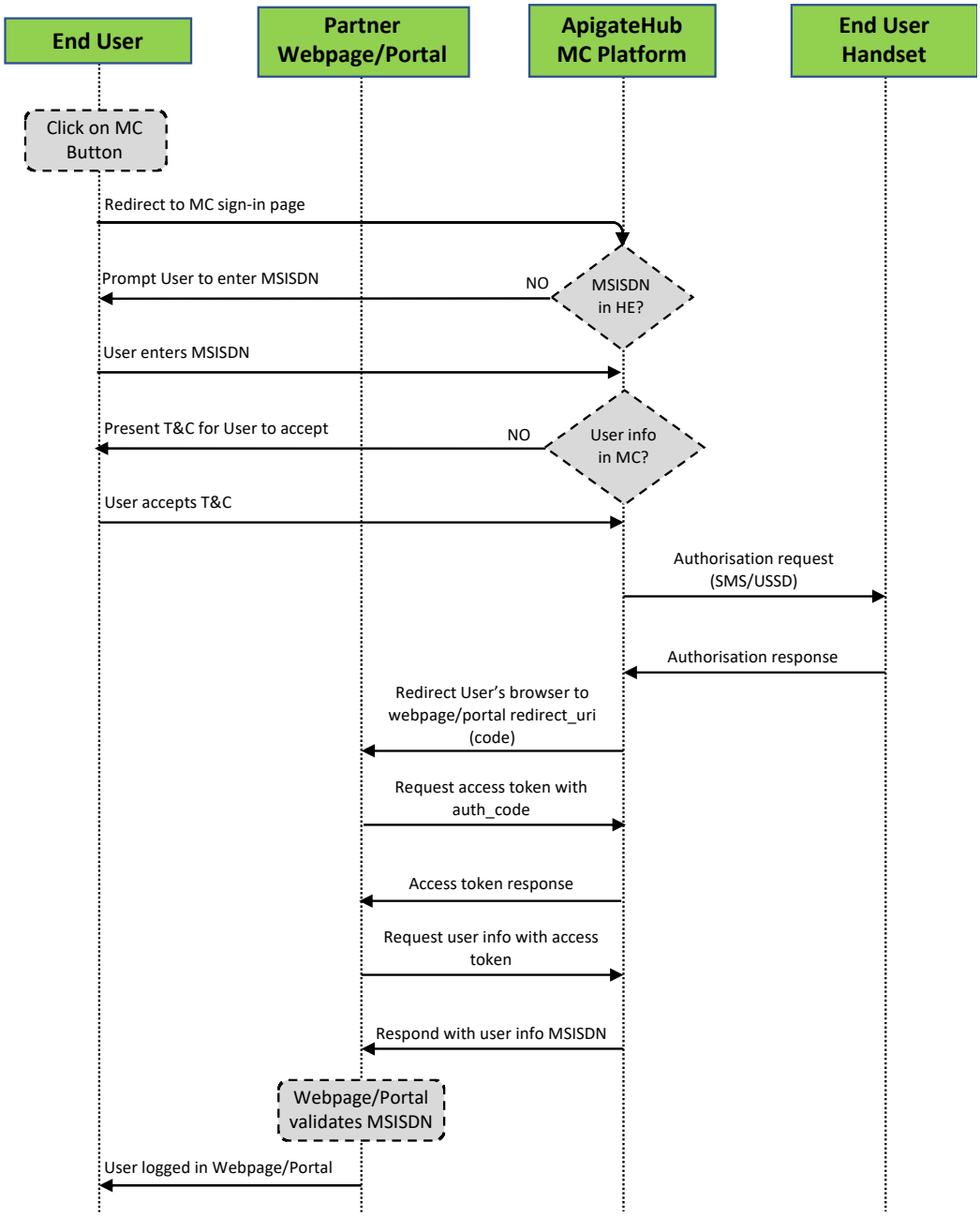
Abbreviation	Definition
API	Application programming interface
SP	Service Provide
MC	Mobile Connect
IDN-LINK	Indentity:Link Service
OIDC	Open ID Connect
HE	Header Enrichment
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
OCS	Online Customer Service
NBWP	Northbound Partner's webpage/portal
SSL	Secure Sockets Layer
TLS	Transport Layer Security
JWT	JSON Web token
T&C	Terms and Conditions

1. Introduction

This document describes Identity:Link (IDN-LINK), also known as Mobile Connect (MC), API specification for Northbound Partners to implement. It lists and explains request and response parameters for each API endpoint.

The remaining of this document is organized as following:
Section 2: Summarizes the MC flow for the Northbound Partner
Section 3: Authorization Request API Details
Section 4: Access Token API Details
Section 5: User info API Details

2. Summary of Mobile Connect flow for Northbound Partner



When user clicks on Mobile Connect button on Northbound Partner's webpage/portal (NBWP), the request is redirected to Mobile Connect sign-in page e.g. using HTML iframe popup. MSISDN is injected as part of a redirections via Header Enrichment if user device is on-net.

Authorization request to user is sent to MSISDN via SMS or USSD. For first time user an authorization request is triggered only after User agrees to login with Mobile Connect terms and conditions.

User agrees to login with Mobile Connect by clicking on a unique link that is contained within the SMS. An authorization code is generated. User's browser is redirected to the NBWP redirect_uri. After this step loading the page is handled by the NBWP.

NBWP calls Token endpoint to request for access token by supplying authorization code. NBWP uses this access token to request for user info such as MSISDN.

3. Authorization request endpoint

Endpoint name: `authorize`
Method: GET
The request parameters are added using Query String Serialization. The following table lists the parameters required for Mobile Connect authorization request.

Parameter	Required	Type (max length)	Description	Value
`response_type`	Mandatory	String(20)	To indicate the grant type flow to be used is authorization code	Must set to `code`
`client_id`	Mandatory	String(50)	SP identifier For authorization request validation	Value will be provided to SP after it is configured in MC server
`scope`	Mandatory	String(20)	List of ASCII strings for authorization scope request	Must set to `openid mc_authz phone`
`redirect_uri`	Mandatory	String(100)	URI where response will be sent to.	SP provide redirect_uri URI in request parameter must match pre-registered redirect_uri configured in MC server
`version`	Mandatory	String(10)		Must set to `2.2`
`state`	Mandatory	String(50)	Security mechanism to	**SP generates unique value.** **This is an alpha-numeric field, at**

			prevent Cross-Site Request Forgery	**least 5 characters in length. The value must be unique for each SP**
acr_values	Mandatory	String(10)	Indicate LoA	2
client_name	Mandatory	String(50)	SP's name	TDB
binding_message	Mandatory	String(45)	Reference ID to be displayed on Consumption device and Authorization device	SP generates unique Reference ID

Table 1: Sign-in request parameters

3.1 SP/Client sends sign-in request to MC server

The below is an example of SP/client sends request to MC server using GET method

Sample request

HTTP GET /authorize?response_type=code
&client_id=OCS_1
&scope=openid%20mc_authz%20phone
&redirect_uri=https%3A%2F%2Fclient.serviceprovider.com.my
&version=2.2
&state=af0oth123
&acr_values=2
&client_name=OCS
&binding_message=REF1134

3.2 SP/Client gets authorization code from MC server

MSISDN is extracted if it is available in Header Enrichment. User is prompted to enter MSISDN if MSISDN is not available. Authorization server authenticates user, gets user consent and returns authorization "code" to SP

The "code" is returned via redirect_uri extracted from request parameters, response parameters are added as query parameter encoded using application/x-www-form-urlencoded.

Parameter	Required	Type (max length)	Description	Value
code	Mandatory	String(50)	Authorization code	code generated as per OAuth 2.0
state	Mandatory	String(50)	Taken from request parameter	Must be same as state submitted in request. If the state does not match the value presented in the initial request, the SP should reject the callback

Table 2: Authorization response parameters

Sample response

```
HTTP 302 Found
Location https://client.serviceprovider.com.my?code=<auth_code>
&state=af0oth123
```

3.3 Authorization Error Response

In case of error e.g. user authorization fails or user does not agree, error will be returned in response. The SP will need to handle the error by appropriately displaying an error page or message.

Parameter	Required	Description	Value
`error`	Mandatory	Error code	See Table 4: List of authorization error code
`error_description`	Mandatory	Human-readable description of error	
`state`	Mandatory	From request parameter	Must be same as state submitted in request

Table 3: Authorization error parameters

error	error_description
invalid_request	Request is missing a parameter, contains an invalid parameter
invalid_client	Invalid client_id is given
access_denied	User denies the request

Table 4: List of authorization error codes

Sample of error response

```
HTTP 302 Found
Location https://client.serviceprovider.com.my?error=access_denied
&error_description= User%20deined%20authorization%20request
&state=af0oth123
```

4. Token endpoint

Token endpoint returns Access token and ID token. Request to token endpoint is sent via HTTPS and encoding is used as application/x-www-form-urlencoded.

Endpoint: `token`
Method: POST

4.1 SP/Client requests for Access Token and ID Token

Request to token endpoint is sent via HTTPS and encoding is used as application/x-www-form-urlencoded.

Parameter	Required	Type (max length)	Description	Value
grant_type	Mandatory	String(45)	Specify the type request	Must set to authorisation_code
code	Mandatory	String(50)	Authorization code	Authorization code obtained from Mobile Connect server
redirect_uri	Mandatory	String(100)	URI where authorization response was sent to.	Must match URI in authorization request
Client credential	Mandatory			The client_id and client_secret used in HTTP Basic Authentication using the OAuth 2.0 Client Password mechanism (RFC 6749 Section 2.3.1 [8]).

Table 5: Token request parameters

Sample request

```
POST /token
Authorisation: Basic T0NTOmhlbGxvd29yZA==
Content-Type: application/x-www-form-urlencoded
grant_type=authorisation_code
&code=<auth_code>
&redirect_uri=https%3A%2F%2Fclient.serviceprovider.com.my
```

Client credential is concatenated string consists of client_id,":", client_secret. The string is encoded using Base64 encoder
For example:
client_id=OCS
client_secret = helloword
Concatenated string = OCS:helloword
Result after encoded with Base64 Encoder T0NTOmhlbGxvd29yZA==

4.2 SP/Client receives Token (Access Token and ID Token)

The response is in accordance with OAuth 2.0

Parameter	Required	Type (max length)	Description	Value
access_token	Mandatory	String(50)	OAuth 2 access token	OAuth 2 access token
token_type	Mandatory	String(20)	For Mobile Connect token_type=bearer	Set to Bearer
id_token	Mandatory	String (600)	Additional token used in OIDC	Refer to ID Token

				parameters
expires_in	Mandatory	Long	Expiration in second from the time token is generated	E.g.300 Token expires in 5 minutes

Table 5: Token response parameters

Sample response

```
HTTP/1.1 200 OK
Content-Type: application/json
{
"access_token":"<access_token>",
"token_type":"Bearer",
"expires_in":300,
"id_token": "<id_token>"
 }
```

ID Token is represented as a JWT (https://tools.ietf.org/html/rfc7519). These are parameters in ID Token

Parameter	Required	Type (max length)	Description	Value
iss	Mandatory	String(100)	Issuer identifier	MC server Domain name
sub	Mandatory	String(50)	Subject identifier	Unique identifier of end user. Case-sensitive ASCII string. Also known as PCR
aud	Mandatory	Long	Intended audience for ID token	SP's client_id
exp	Mandatory	Long	Token expiry time after which ID Token must not be accepted for processing	Unix timestamp. Number of second since 1970-01-01 E.g 1500193668
iat	Mandatory	Long	The time ID Token was issued	Unix timestamp. Number of second since 1970-01-01 E.g 1500193668
acr	Mandatory	String(10)	Authentication context class	Fixed value: 2
amr	Mandatory	String(10)	Authentication methods reference	Fixed value: SMS

Table 6: ID Token parameters

Sample of an unsigned ID Token raw JSON

```
{
  "iss": "https://mobileconnect.example.com",
  "sub": "awxtf56874axs",
  "aud": "OCS",
  "exp": 1500194508,
  "iat": 1500194491,
  "acr": 2,
  "amr": "SMS"
```

```
}
```

Raw JSON value will be signed using JSON signature, made into JWT and returned as part of Token response.

4.3 Access token error response

Token endpoint will respond with code and description in case of exception or error

error	error_description
`invalid_request`	Request is missing a parameter, contains an invalid parameter
`invalid_client`	Invalid client_id is given
`invalid_grant`	The authorization code is invalid. This is also the error if the redirect URI given in the authorization does not match the URI provided in this access token request.

Table 7: Access token error codes

Sample of error response

```
HTTP/1.1 400 Bad Request
Content-Type: application/json;charset=UTF-8
{
  "error": " invalid_grant ",
  "error_description": "Invalid authorization code"
}
```

5. User info endpoint

The SP/Client accesses the UserInfo endpoint using the Access Token returned to it. Server will validate access token and return
Method: GET
Endpoint: `userinfo`

5.1 SP/Client requests for User Info

Sample request

```
Get  /userinfo
Accept: application/x-www-form-urlencoded
Authorisation: Bearer <access_token>
```

5.2 SP/Client gets user info response

List of user info response parameters

Parameter	Required	Description	Value
`sub`	Mandatory	Subject identifier	Unique identifier of end user. Case-

			sensitive ASCII string. PCR
phone_number	Optional	Contains user phone number just only for NBWP	Phone number in E. 164 format, including international prefix .e.g. +60 for Malaysia

Table 8: User info response

Sample of user info response

```
HTTP/1.1 200 OK
Content-Type: application/json
{
    "sub": "awxtf56874axs",
    "phone_number": "+6012345678"
}
```

For invalid token User Info endpoint responds with HTTP 401 and error message. Sample of error response

```
HTTP/1.1 401 Unauthorized
Content-Type: application/json
{
    "error": " invalid_token",
    "error_description":"userinfo request was made with invalid token”
}
```

6. List of error codes and descriptions

The following table lists error codes and their descriptions

Error code	Description
access_denied	User denies the authorization request
invalid_request	Request is missing a parameter, contains an invalid parameter
invalid_client	Invalid client_id is given
invalid_grant	The authorization code is invalid. This is also the error if the redirect URI given in the authorization does not match the URI provided in this access token request.
invalid_token	Request was made with invalid token

Table 9: Error codes and descriptions