Mobile Connect Technical Reference
Version 2.0
11 August 2017

This is a Non-binding Permanent Reference Document of the GSMA

Security Classification: Non-confidential

Copyright Notice

Disclaimer

Antitrust Notice

Table of Contents

1 Introduction

1.1 Overview

The GSMA Identity programme focuses on positioning Operators as trusted providers of identity and attribute services to third party Service Providers. Within this, the programme identifies a set of Authentication, Authorisation and Identity & Attribute products that collectively are referred to as Mobile Connect.

This document specifies:

- **Error handling mechanism** for both device-initiated and server-initiated Mobile Connect services
- **Error messages**: error messages and example descriptive text to be displayed to the user
- **Identity Services attribute list**: normative claim names for Identity Services attributes.
- **Mobile Connect Attribute services** normative claims names for attribute services [i.e. Mobile Connect Know Your Customer (MC KYC), Mobile Connect Account Takeover Protection (MC ATP) and Mobile Connect Verified MSISDN (MC VM).

1.2 Definitions

Term	Description
Mobile Connect Authentication	Provides single factor and two-factor authentication using the mobile phone as the authentication device.
Mobile Connect Authorisation	Provides single factor and two-factor authorisation using the mobile phone as the authorisation device. It also supported the first party and 3rd party authorisation.
Mobile Connect Identity Services	Retrieve Mobile Connect user's personal data after a successful consent capture.
Mobile Connect Attribute services	Retrieve specific attributes of the Mobile Connect User, Account related claims or result of operations.

1.3 Abbreviations

Term	Description
RFC	Request for Comments
SP	Service Provider
IOT	Internet Of Things
MCIS	Mobile Connect Identity Services
OIDC	OpenID Connect

1.4 References

Ref	Doc Number	Title
[1]	RFC 2119	"Keywords for use in RFCs to Indicate Requirement Levels," S. Bradner, March 1997. Available at http://www.ietf.org/rfc2119.txt
[2]	RFC 2616	"Hypertext Transfer Protocol (HTTP) an application level protocol," J Gettys, J. Mogul, L. Masinter, P. Leach, T. Berners-Leem June 1999. Available at http://www.ietf.org/rfc/rfc2616.txt
[3]	RFC 6749	"The Oauth 2.0 Authorization Framework," D. Hard5, Ed. October 2012 available at http://www.ietf.org/rfc/rfc6749.txt
[4]	OpenID Connect	"An interoperable authentication protocol based on the Oauth 2.0 family of specifications" available at http://openid.net/developers/specs/
[5]	OpenID Discovery	OpenID Connect discovery specifications. https://openid.net/specs/openid-connect-discovery-1_0.html

1.5 Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECMMENDED", "NOT RECMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1].

1.6 Technical documentation map

The Mobile Connect architecture, technical specifications and implementation guidelines are encompassed by a set of documentation as laid out below:

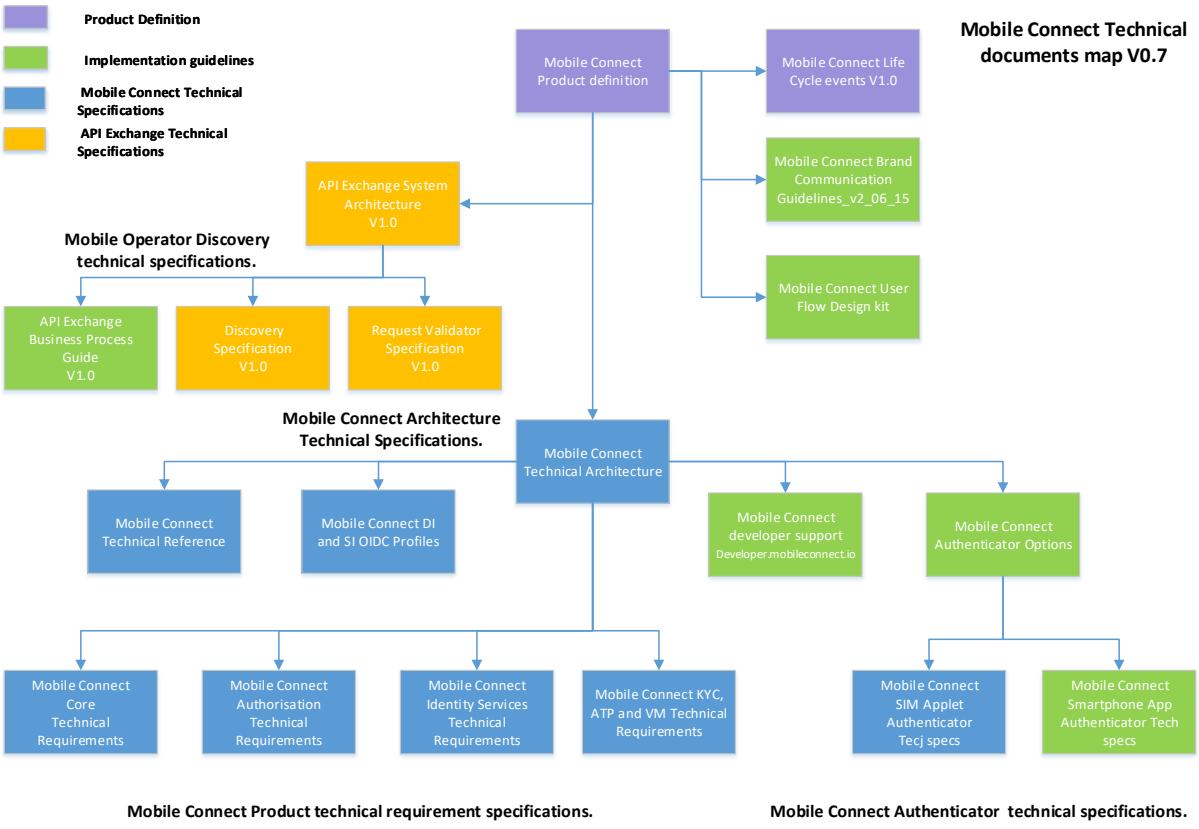

Figure 1 : Mobile Connect technical documentation map

2 Scope Reference

The following scope values are valid for use in the OIDC request parameter “scope”.

2.1 Mobile Connect Authentication scopes

The following table defines the scopes that SHOULD be used for Mobile Connect Authentication products. A release 2 ID GW implementation must support both values.

Scope	Description
“openid”	By default, Mobile Connect authentication must be executed.
“openid mc_authn”	Same as above, but only supported from Mobile Connect Release 2 onwards.

Table 1: Mobile Connect authentication scopes

2.2 Mobile Connect Authorisation scopes

The following table defines the scopes that SHOULD be used for Mobile Connect Authorisation products.

Scope	Description
“openid mc_authz”	OIDC scope to identity Mobile Connect Authorisation products.

Table 2: Mobile Connect authorisation scope

2.3 Mobile Connect Identity Services scopes

The following table describes the scopes that SHOULD be used within Mobile Connect Identity Service products.

Scope	Description
“openid mc_identity_phonenumber”	Execute authentication and capture consent for MC enabled phone number
“openid mc_identity_signup”	Execute authentication and capture consent for sign up details
“openid mc_identity_nationalid”	Execute authentication and capture consent for national ID.

Table 3: Mobile Connect Identity services scopes

2.4 Mobile Connect Attribute Services Scopes

The following table describes the scopes that SHOULD be used within Mobile Connect attribute services.

Scope	Description
`"openid mc_atp"`	Execute authentication and capture consent to share account call settings, sim status and device change.
`"openid mc_kyc_plain"`	Based on IDGW policies, if required execute authentication and capture consent to compare the SP provided customer information with the Mobile Connect User's information held by the Operator. The SP provides attributes in plain text format.
`"openid mc_kyc_hashed"`	Based on ID GW policies, if required execute authentication and capture consent to compare the SP provided customer information with the Mobile Connect User's information held by the Operator. The SP provides attributes in hashed format.
`"openid mc_attr_vm_share"`	Retrieve the device MSISDN from Operator's network. It works in the On-net scenario only.
`"openid mc_attr_vm_match"`	Compare the SP provided MSISDN with the device MSISDN, and return the result in Boolean format. The SP provides the attribute values, in a plain text format.
`"openid mc_attr_vm_match_hash"`	Compare the SP provided MSISDN with the device MSISDN, and return the result in Boolean format. The SP provides the attribute values, in a plain text format.

Table 4: MC Attribute Service Scope values

3 OIDC Response Amr Values

The following table defines the OIDC response `amr` value that must be returned by an ID GW, for the most commonly used authenticators across the different Mobile Connect products.

Amr claim value	Mobile Connect Authentication	Mobile Connect Authorisation	Description
SIM_OK	Mobile Connect Authenticate (single factor)	Mobile Connect Authorise (single factor)	SIM Applet authenticator , Uses SIM as a secure element and secure execution environment to support single-factor Mobile Connect authentication / authorisation
SIM_PIN	Mobile Connect Authenticate Plus (two factor)	Mobile Connect Authorise Plus (two factor)	SIM Applet authenticator, Uses SIM as a secure element and secure execution environment to support Two-Factor Mobile Connect authentication / authorisation
SM_APP_OK	Mobile Connect Authenticate (single factor)	Mobile Connect Authorise (single factor)	Smartphone app authenticator, to support single-factor Mobile Connect authentication / authorisation.
SM_APP_PIN	Mobile Connect Authenticate Plus (two factor)	Mobile Connect Authorise Plus (two factor)	Smartphone app authenticator, to support Two-factor Mobile Connect authentication / authorisation.
USSD_OK	Mobile Connect Authenticate (single factor)	Mobile Connect Authorise (single factor)	Network initiated USSD based authenticator, to support Mobile Connect single-factor authentication/ authorisation.
USSD_PIN	Mobile Connect Authenticate Plus (two factor)	Mobile Connect Authorise Plus (two factor)	Network initiated USSD based authenticator, to support Mobile Connect Two-factor authentication/ authorisation.
SMS_URL_OK	Mobile Connect Authenticate (single factor)	Mobile Connect Authorise (single factor)	SMS URL based authenticator, to support Mobile Connect single-factor authentication/ authorisation.
SEAM_OK	Mobile Connect Authenticate (single factor)	Not Available	Seamless authenticator to support Mobile Connect Single-factor authentication only. For Mobile Connect Authorisation and Mobile Connect Identity Services it MUST not be supported.

Table 5 : amr values

4 Error Handling Mechanism

Mobile Connect services can be offered either in device-initiated or server-initiated modes. Error messages must be propagated differently in both the scenarios.

4.1 Error Handling in Device Initiated Mode

Mobile Connect follows the OpenID Connect error handling mechanism to send any errors back to the Service Provider. These errors can be returned in a query string to the Service Provider on `redirect_uri` [through HTTP redirect 302] when the redirect URI is valid, or

HTTP 400 Bad Request with a human readable string OR JSON object [Implementation choice] containing the error code and error description when the redirect URI is invalid. Errors must be returned as a JSON object containing the error code and error description using appropriate HTTP status codes for token and resource endpoints. These errors are returned to the Service Provider as described in the OpenID Connect specifications:

- https://openid.net/specs/openid-connect-core-1_0-17.html#AuthError
- https://openid.net/specs/openid-connect-core-1_0-17.html#TokenErrorResponse
- https://openid.net/specs/openid-connect-core-1_0-17.html#UserInfoError [applicable to service specific resource endpoint]

4.2 Error Handling in Server Initiated Mode

Mobile Connect must return the errors to the Service Provider using HTTP status codes and a JSON object. The errors can be returned in the following scenarios.

- Mobile connect server-initiated authorization endpoint
- Mobile Connect Push Notification [i.e. errors through notification]
- Errors to Operator IDGW through Acknowledgement [from Service Provider]
- https://openid.net/specs/openid-connect-core-1_0-17.html#UserInfoError [applicable to service specific resource endpoint]

5 Error Messages

Mobile Connect defines various error messages based on different scenarios that will be returned to the Service Provider and can receive errors from the Service Provider. These error messages are related to:

- Missing MANDATORY parameter(s) in authorization request
- Incorrect parameter values in authorization request
- Incorrect token request
- Incorrect Notification
- Errors to IDGW through acknowledgement to notification
- Errors mapped to abstract errors
- Mobile Connect User interaction failure
- Mobile Connect Service/Product specific errors

5.1 Generic Errors

This section describes Mobile Connect generic errors applicable to all services.

5.1.1 Mobile Connect in Device Initiated Mode

Operators can offer Mobile Connect services in device-initiated mode where service is initiated through user-agent [i.e. native app, web browser etc.,]. Error responses are returned from authorize, token and resource endpoints as described below.

5.1.1.1 Error Responses: Device Initiated Authorize Endpoint

If authorization request contains a `state` parameter, then the error response MUST contain "`state` "parameter value. The value is set to the value received from the Service Provider. If `correlation_id` is provided in the authorization request, then it must be included in the response, and the value must be set to the value received from the Service Provider.

The following is the example error response.

```
HTTP/1.1 302 Found
Location: https://sp.example.org/redirct_here?
 error=invalid_request
 &error_description=Invalid%20response_type%20value
 &state=af0ifjsldkj
 &correlation_id=<example correlationid>
```

If `redirect_uri` is missing or `redirect_uri` value is invalid, then Operator IDGW always treat this as a high priority error and MUST return the appropriate error response as described in the following table.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
MSISDN/ENCR MSISDN/ PCR provided does not belong to the Operator.	Redirect 302	access_denied	"Unknown user" [OR] "User is not recognized"
MSISDN/ENCR_MSDISDN belongs to the Operator, but MC services are not enabled, Note: IDGW policy does not allow to create partially active account	Redirect 302	access_denied	"Mobile Connect User is not registered" [OR] "Unknown Mobile Connect User".
System connection problems [internal to IDGW] (or) Authenticator unreachable (or) Expiration in server (or) Any Unexpected error [internal to IDGW]	Redirect 302	server_error (or) temporarily_unavailable	Internal Server Error.
Multiple requests for the same MSISDN sent at the same time	Redirect 302	access_denied	The user is busy with another transaction.
redirect_uri value is invalid or not a registered URI.	Bad Request 400	invalid_request	redirect_uri is invalid.
response_type parameter is missing or invalid	Redirect 302	invalid_request	MANDATORY parameter response_type is missing or value is invalid.
client_id parameter is missing	Redirect 302	invalid_request (or) access_denied	MANDATORY parameter client_id is missing
client_id parameter value is invalid	Redirect 302	invalid_request (or) unauthorized_client (or) access_denied	The client is not authorized to request an authorization code.
client_id is valid, but not allowed to make MC service requests	Redirect 302	access_denied (or) unauthorized_client	The client is not allowed to make MC service requests.
scope parameter is missing (or) scope value "openid" is missing (or) invalid scope values (ex: "abcd")	Redirect 302	invalid_request (or) invalid_scope	MANDATORY parameter scope is missing (or) invalid scope value
version parameter is missing (or) version parameter value is invalid	Redirect 302	invalid_request	MANDATORY parameter version is missing / invalid.
state parameter exists, but the value is invalid	Redirect 302	invalid_request	RECOMMENDED parameter state is invalid
nonce parameter is missing or nonce parameter exists, but the value is empty	Redirect 302	invalid_request	MANDATORY parameter nonce is missing (or) invalid.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`login_hint` and `login_hint_token` parameters are missing Note: In the following conditions the IDGW SHOULD prompt the user to input MSISDN instead of throwing an error; -- MC service is for stand-alone authentication only in the device-initiated mode. -- If IDGW policy allows to capture MSISDN directly from the user [FC mode only].	Redirect 302	`invalid_request`	MANDATORY parameters `login_hint_token` (or) `login_hint` does not exist.
`login_hint` and `login_hint_token` both exist	Redirect 302	`invalid_request`	Malformed request, duplicate parameter entries
`login_hint` (or) `login_hint_token` value is invalid	Redirect 302	`invalid_request`	Invalid value for `login_hint` or `login_hint_token`
`acr_values` parameter is missing (or) `acr_values` exist but contains an invalid value, other than supported values	Redirect 302	`invalid_request`	MANDATORY parameter acr_values are missing or invalid values.
`display` parameter exists and it has an invalid value (or) `display` parameter exists and IDGW does not support the requested value	Redirect 302	`invalid_request`	Invalid display value. / not supported.
The same parameter exists multiple times	Redirect 302	`invalid_request`	Multiple parameter names in the authorization request. Malformed request.
`prompt` value exists, and it has an invalid value	Redirect 302	`invalid_request`	`prompt` value is invalid
`claims` parameter exists, but it does not contain any value (or) have invalid values.	Redirect 302	`invalid_request`	`claims` value is invalid,
GET request is used, and the request parameter is NOT serialized using URI string serialization, IDGW able to validate the `redirect_uri`.	Redirect 302	`invalid_request`	GET request invalid serialization
POST request is used, the request parameters are NOT serialized using form serialization, IDGW can validate the redirect_uri	Redirect 302	`invalid_request`	POST request Invalid serialization

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`response_mode` parameter exists, and it has a value not supported by the Operator IDGW	Redirect 302	`invalid_request`	`response_mode` contains same as `response_type` (or) invalid.
`max_age` parameter exists, and it has an invalid value	Redirect 302	`Invalid_request`	Invalid `max_age` value
Multiple problems in authorization request [redirect URI is valid]	Redirect 302	`invalid_request`	Malformed request multiple problems exist
`correlation_id` exists, but it has an empty value	Redirect 302	`invalid_request`	Invalid correlation_id value.
`client_name` exists but it has empty value (or) `client_name` parameter exists, but the provided value is not a registered `client_name` with Operator IDGW and invalid	Redirect 302	`invalid_request`	Invalid `client_name` value

Table 6 : MC Services: Generic Errors-Device Initiated Authorize Endpoint

5.1.1.2 Error Responses: Device Initiated Token Endpoint

The token request is always a server-initiated request. It must be a POST request. An SP makes a token request by presenting the parameters and the form serialization to the Token endpoint. After validating the request, the token endpoint must return the errors in the following format:

```
HTTP/1.1 400 Bad Request
  Content-Type: application/json
  Cache-Control: no-store
  Pragma: no-cache

  {
   "correlation_id": "<example correlation id value>",
   "error": "invalid_request",
   "error_description": "mandatory parmeter is missing"
  }
```

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`grant_type` parameter is missing (or) `grant_type` parameter exists, but value is invalid or not supported	Bad Request 400	`invalid_request (or)` `unsupported_grant_type`	MANDATORY parameter `grant_type` is missing (or) invalid
Authorization code parameter is missing (or) Code parameter exists but value is invalid (or) Code parameter exists, value is already used or expired (or) Authorization code is valid, but not issued to the authenticated client (or) Authorization code is valid, but it is not issued to a MC OpenID Connect request	Bad Request 400	`invalid_grant (or)` `invalid_request`	MANDATORY parameter code is missing (or) invalid (or) expired
`redirect_uri` parameter is missing (or) `redirect_uri` parameter exist and it has a value that DOES NOT match the one sent in the authorization request (or) `redirect_uri` parameter exists, and it has unregistered value (or) Operator IDGW has multiple redirect URI values registered for a given client_id. Redirect parameter exist and it has a value that matches one of the redirect URI registered with Operator IDGW, but the value DOES NOT match the one sent in the previous authorization request	Bad request 400	`invalid_request`	MANDATORY parameter `redirect_uri` is missing (or) is invalid
`client_id` parameter does not exist (or) `client_id` parameter exists, but it has a value that is not registered at Operator IDGW	Bad Request 400 (or) 401	`access_denied (or)` `invalid_client`	Invalid client credentials

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`client_secret` parameter does not exist (or) `client_secret` parameter exists, but it has invalid value (or)	Bad Request 400 (or) 401	`access_denied (or) invalid_client`	Invalid client credentials
`correlation_id` does not exist but previous authorization request and response has correlation ID (or) `correlation_id` exists but it has empty value (or) `correlation_id` exists but it has value that DOES NOT match the one sent in the previous authorization request	Bad Request 400	`invalid_request`	Missing MANDATORY parameter correlation ID (or) invalid
Same parameter exists multiple times	Bad Request 400	`invalid_request`	Malformed request, the same parameter exists multiple times
Unexpected error	Internal server Problem 500	`server_error`	Internal error,
System connection problem	Service Unavailable 503	`server_error`	Service is not available,
SP sends token request through POST, but without form serialization	Bad Request 400	`invalid_request`	No form serialization exists
Multiple problems in token request	Bad Request 400	`access_denied`	Multiple problems were in the token request.

Table 7 : MC Services: Generic Errors - Device Initiated Token Endpoint

5.1.2 Mobile Connect in Server Initiated Mode

The Operator can offer Mobile Connect services in server-initiated mode where service is initiated from a Server. Error responses are returned from authorize endpoint, push notification service and resource endpoints as described below.

5.1.2.1 Error Responses: Server Initiated Authorize Endpoint

For Mobile Connect server-initiated the authorization request must contain a signed request object. This section describes the error format and error messages returned from server-initiated authorize endpoint.

The following is the example error response.

```
HTTP/1.1 400 Bad Request
Content-Type: application/json
Cache-Control: no-store
Pragma: no-cache

{
 "correlation_id": "<example correlation id value>",
 "error": "invalid_request",
 "error_description": "mandatory parameter scope is missing"
}
```

5.1.2.1.1 Error Responses: Request Object Parameter Validation

The following tables describe all request object parameters; their validation and corresponding error responses.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
MSISDN/ENCR_MSISDN/ PCR provided does not belong to the Operator.	Bad Request 400	access_denied	Mobile Connect User is not recognized. [OR] Unknown User
MSISDN/ENCR_MSDISDN belongs to the Operator, but MC services are not enabled, Note: IDGW policy does not allow to create partially active account	Bad Request 400	access_denied	"Mobile Connect User is not registered" [OR] "Unknown Mobile Connect User".
response_type parameter is missing or invalid	Bad Request 400	invalid_request	MANDATORY parameter response_type is missing or value is invalid.
client_id parameter is missing	Bad Request 400	invalid_request (or) access_denied	MANDATORY parameter client_id is missing
client_id parameter value is invalid	Bad Request 400	invalid_request (or) unauthorized_client (or) access_denied	The client is not authorized to request an authorization code.
client_id is valid, but not allowed to make MC service requests	Bad Request 400	access_denied (or) unauthorized_client	The Client is not allowed to make MC service requests.
scope parameter is missing (or) scope value "openid" is missing (or) invalid scope values (ex: "abcd")	Bad Request 400	invalid_request (or) invalid_scope	MANDATORY parameter scope is missing (or) invalid scope value
version parameter is missing (or) version parameter value is invalid	Bad Request 400	invalid_request	MANDATORY parameter version is missing (or) invalid.
nonce parameter is missing or nonce parameter exists but value is empty	Bad Request 400	invalid_request	MANDATORY parameter nonce is missing (or) invalid.
login_hint and login_hint_token parameters are missing	Bad Request 400	invalid_request	MANDATORY parameters login_hint_token (or) login_hint does not exist.
login_hint and login_hint_token both exist	Bad Request 400	invalid_request	Malformed request, duplicate parameter entries
login_hint (or) login_hint_token value is invalid	Bad Request 400	invalid_request	Invalid value for login_hint (or) login_hint_token
acr_values parameter is missing (or) acr_values exist but contains invalid value, other than supported values	Bad Request 400	invalid_request	MANDATORY parameter acr_values is missing (or) invalid values.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Same parameter exists multiple times	Bad Request 400	`invalid_request`	Multiple parameter names in the authorization request. Malformed request.
`claims` parameter exists but it does not have any value (or) invalid values.	Bad Request 400	`invalid_request`	`claims` value is invalid,
GET request is used, and the request parameters are NOT serialized using URI string serialization, IDGW able to validate the `redirect_uri`.	Bad Request 400	`invalid_request`	GET request invalid serialization
POST request is used, the request parameters are NOT serialized using form serialization, IDGW can validate the redirect_uri	Bad Request 400	`invalid_request`	POST request Invalid serialization
`response_mode` parameter exists, and it has a value not supported by the Operator IDGW	Bad Request 400	`invalid_request`	`response_mode` contains same as `response_type` (or) invalid.
`max_age` parameter exists, and it has an invalid value	Bad Request 400	`invalid_request`	Invalid `max_age` value
Multiple problems in authorization request [redirect URI is valid]	Bad Request 400	`invalid_request`	Malformed request multiple problems exist
`correlation_id` exists, but it has empty value	Bad Request 400	`invalid_request`	Invalid correlation_id value.
`client_name` exists, but it has empty value (or) `client_name` parameter exists, but the provided value is not a registered `client_name` with Operator IDGW and is invalid	Bad Request 400	`invalid_request`	Invalid `client_name` value
`client_notification_token` MANDATORY parameter is missing (or) `client_notification_token` exists, but it has an invalid value	Bad Request 400	`invalid_request`	MANDATORY parameter client_notification_token is missing (or) invalid
`notification_uri` parameter is missing (or) `notification_uri` exists but it is not registered with IDGW (or) has an invalid value	Bad Request 400	`invalid_request`	MANDATORY parameter `notification_uri` is missing (or) invalid.
`iss` parameter is missing (or) `iss` parameter exists, but it has an invalid value	Bad Request 400	`invalid_request`	MANDATORY parameter SP's `iss` parameter is missing (or) invalid
`aud` parameter is missing (or) `aud` parameter exists, but it has an invalid value	Bad Request 400	`invalid_request`	`aud` parameter is missing (or) invalid.

Table 8 : MC Services: Generic Errors - Server Initiated Request Object Validation

5.1.2.1.2 Error Responses: Server Initiated Authorization Request

The following table describes the MC Authorization Request parameters which includes request object.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`response_type` parameter is missing (or) `response_type` parameter exists value is invalid (or) `response_type` parameter exists and the value is valid, but not matching the request object `response_type` value.	Bad Request 400	`invalid_request`	MANDATORY parameter `response_type` is missing (or) invalid (or) malformed request; `response_type` values does not match
`client_id` parameter does not exist	Bad Request 400	`access_denied (or) invalid_request`	MANDATORY parameter client ID does not exist
`client_id` parameter exists, but it has invalid value	Bad Request 400	`access_denied (or) unauthorized_client`	Unknown client ID
`client_id` parameter exists, but the value does not match the request object `client_id` value	Bad Request 400	`invalid_request`	Malformed request, ambiguous client ID values
`scope` parameter is missing (or) `scope` parameter exists, it does not contain "openid"	Bad Request 400	`invalid_request`	MANDATORY parameter scope parameter is missing
`scope` parameter exists, but it does not match the request object scope value	Bad Request 400	`invalid_request`	Malformed request, ambiguous scope values
`scope` parameter exists, request scope is not supported by the Operator IDGW	Service Unavailable 503	`server_error`	Service is not available.
Request object parameter does not exist (or) request object parameter exists, but value is not valid	Bad Request 400	`invalid_request`	MANDATORY parameter request is missing
Signature validation of request object is failed	Bad Request 400	`invalid_request`	Malformed request, invalid signature
System connection problem (or) Expiration in server	Service Unavailable 503	`server_error`	Service is not available,
Multiple requests for the same MSISDN sent at the same time	Internal Server Error 500	`server_error`	The user is busy with another transaction.
Unexpected error [Internal to IDGW]	Internal Server Error 500	`server_error`	Internal Server Error

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
IDGW time-out due to internal error.	Internal Server Error 500	server_error	Timeout: Server internal error.

Table 9 : MC Services: Generic Errors – Server Initiated Authorize Request Validation

5.1.2.2 Error Response: Notification [IDGW to SP]

The following table describes the generic errors that can be returned to SP's notification endpoint from Operator IDGW.

The error format will be like the following:

```
POST /token_notif HTTP/1.1
Content -Type: application/json
Host: spserver.example.com
Authorization: bearer Bedsfe2134sd
{
   "correlation_id": "42da5b19-457a-4d30-a5c4-038c62dccbb0",
   "auth_req_id": "1234455",
   "error": "access_denied",
   "error_description": "authentication failure example"
```

Error Scenario	Error code	Error Description [RECOMMENDED text]
Authenticator unreachable/ expiration in server	server_error	Service is not available,
System connection problems [internal to IDGW]	server_error	Connection problem.
Unexpected error [Internal to IDGW]	server_error	Internal server error.

Table 10 : MC Services: Generic Errors - Server Initiated Notification IDGW to SP

5.1.2.3 Error Responses: ACK To Notification [SP to Operator IDGW]

This section describes the error responses from SP to Operator IDGW through acknowledgement.

The Error format is like the following.

```
HTTP/1.1 400 Bad Request
  Content-Type: application/json
  Cache-Control: no-store
  Pragma: no-cache

  {
   "correlation_id": "42da5b19-457a-4d30-a5c4-038c62dccbb0",
   "error": "invalid_request",
   "error_description": "invalid tokens"
  }
```

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Invalid JWT received from Operator IDGW [ID Token]	Bad Request 400	invalid_request	Invalid JWT [ID Token] is received.
Invalid access_token is received	Bad Request 400	invalid_request	Invalid access_token is received
Invalid auth_req_id	Bad Request 400	invalid_request	Malformed request, unable to identify the response.
Invalid correlation_id	Bad Request 400	invalid_request	Malformed request, unable to correlate the response.
token_type parameter is missing	Bad Request 400	invalid_request	MANDATORY parameter token_type is missing
token_type parameter exists, but the value is not Bearer	Bad Request 400	invalid_request	Invalid token type value
expires_in parameter does not exist	Bad Request 400	invalid_request	MANDATORY parameter expires_in does not exist.
expires_in parameter exists, but value is invalid	Bad request 400	invalid_request	Invalid expires_in value
ID Token parameter does not exist	Bad Request 400	invalid_request	MANDATORY parameter ID Token does not exist
Access Token parameter does not exist	Bad Request 400	invalid_request	MANDATORY parameter access token does not exist
System connection problem (or) Expiration in server	Service Unavailable 503	server_error	Service is not available,
Unexpected error [Internal to SP]	Internal Server Error 500	server_error	Internal Server Error
SP server time-out due to an internal error.	Internal Server Error 500	server_error	Timeout: Server internal error.

Table 11 : MC Services: Generic Errors - Server Initiated ACK SP to IDGW

5.1.3 Error Responses: Resource Endpoint

There are two different types of resource endpoints for attributes in Mobile Connect. One is to share Mobile Connect User attributes through PremiumInfo endpoint and another one is Service specific resource endpoint. This section describes generic errors that can be returned from both the endpoints.

5.1.3.1 Error Responses Service Specific/ PremiumInfo Resource Endpoint

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Unexpected error	Internal Server Error 500	`server_error`	Internal server error,
System connection problems	Service Unavailable 503	`server_error`	Service is not available,
Access token is sent but is not obtained through Mobile Connect OIDC request (or) Access token exists, but it is invalid (or) (or) the expired access token	Unauthorized 401	`invalid_request (or) invalid_token`	Invalid access token (or) access token does not exist (or) expired invalid access token
Resource request is sent using POST and "`access_token`" parameter does not exist in the Form encoded body	Unauthorized 401	If the access token does not exist, then the following error SHOULD be returned. Error code and error description must not be returned. Example: `HTTP/1.1 401 Unauthorized WWW-Authenticate: Bearer`	
The resource request is sent through POST entity-header includes the "Content-type header and the value is NOT "application/x-www-form-urlencoded".	Bad Request 400	`invalid_request`	
The resource request is sent through POST and it is not form url encoded as described in RFC 6750	Bad Request 400	`invalid_request`	Malformed request, invalid url encoding
The resource request is sent through POST, and the content to be encoded in the entity-body contains non-ASCII characters as defined In RFC 6750	Bad Request 400	`invalid_request`	Malformed request, invalid non-ascii characters
Any unsupported parameters exist in the request	Bad Request 400	`invalid_request`	Malformed request, invalid parameters
Multiple problems in the resource request	Bad Request 400	`invalid_request`	Malformed request, invalid parameters.
The request requires higher privileges than provided by the access token	Forbidden 403	`insufficient_scope (or) access_denied`	Insufficient scope.
Unexpected error [Internal to Resource Server]	Internal Server Error 500	`server_error`	Internal Server Error
Resource server time-out due to internal error.	Internal Server Error 500	`server_error`	Timeout: Server internal error.

Table 12 : MC Services: Generic Errors - Resource Endpoint

5.2 Mobile Connect Core & Authentication

5.2.1 Mobile Connect in Device Initiated Mode

The Operator can offer Mobile Connect Authentication service in the device-initiated mode where service is initiated through user-agent [i.e. native app, web browser, etc.,]. Error responses are returned from authorize endpoint, token and resource endpoints as described below.

5.2.1.1 Error Responses: Device Initiated Authorize Endpoint

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Mobile Connect User failed to authenticate [example: invalid pin]	Redirect 302	`authentication_failure` [or] `access_denied`	Mobile Connect user failed to authenticate
Mobile Connect User cancelled or rejected the authentication request on his/her mobile device	Redirect 302	`authentication_denied` (or) `authentication_failure` (or) `access_denied`	Mobile Connect user rejected / cancelled the authentication
Mobile Connect User unable to authenticate timeout occurred	Redirect 302	`authentication_failure` (or) `access_denied`	Timeout occurred during authentication.

Table 13 : MC Authentication: Errors - Device Initiated Authorize endpoint

5.2.1.2 Error Responses: Device Initiated Token Endpoint

None. Already covered in the generic errors section.

5.2.2 Mobile Connect in Server Initiated Mode

The Operator can offer Mobile Connect services in server-initiated mode where service is initiated from a Server. Error responses are returned from authorize endpoint, push notification service and resource endpoints as described below.

5.2.2.1 Error Responses: Server Initiated Authorize Endpoint

This section describes service specific error responses in the server-initiated model.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Requested authentication strength is not implemented.	Bad Request 400	`invalid_request`	Requested authentication is not supported.
Authentication for the given strength is implemented, but service is not available now due to internal error.	Service Unavailable 503	`server_error`	Request authentication is temporarily not available.

Table 14 : MC Authentication: Errors - Server Initiated Authorize Endpoint

5.2.2.2 Error Responses: Notification [IDGW to SP]

Error Scenario	Error code	Error Description [RECOMMENDED text]
Mobile Connect User failed to authenticate	`authentication_failure` (or) `access_denied`	Mobile Connect user failed to authenticate
Mobile Connect User cancelled or rejected the authentication request on his/her mobile device	`authentication_denied` (or) `authentication_failure` (or) `access_denied`	Mobile Connect user rejected / cancelled the authentication
Mobile Connect User is prompted with authentication, the timeout occurs.	`authentication_failure` (or) `access_denied`	Timeout occurred during authentication.

Table 15: MC Authentication: Errors - Server Initiated Notification IDGW to SP

5.2.2.3 Error Responses: ACK To Notification [SP to IDGW]

Error Scenario	HTTP mode	Error Code	Error Description [RECOMMENDED text]
Invalid authentication proof token [ID Token]	Bad Request 400	`invalid_request`	Mobile Connect Authentication proof token [ID token] is invalid.

Table 16 : MC Authentication: Errors – Server Initiated ACK SP to IDGW

5.3 Mobile Connect Authorisation

5.3.1 Mobile Connect in Device Initiated Mode

The Operator can offer Mobile Connect Authorisation service in the device-initiated mode where service is initiated through user-agent [i.e. native app, web browser, etc.,]. Error responses are returned from authorize, token and resource endpoints as described below.

5.3.1.1 Error Responses: Device Initiated Authorize Endpoint

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
In the single-page environment Mobile Connect User failed to approve the requested prompt (or) IDGW unable to assert the user's identity	Redirect 302	`authorisation_failure` (or) `access_denied`	Mobile Connect user failed to approve the requested `prompt`
In the single-page environment Mobile Connect User cancelled or rejected the MC authorisation request on his/her mobile device	Redirect 302	`authorisation_denied` (or) `authorization_failure` (or) `access_denied`	Mobile Connect user rejected / cancelled the authentication
Mobile Connect User unable to authorise timeout occurred	Redirect 302	`authorisation_failure` (or) `access_denied`	Timeout: User is not available to respond, later.
Requested authorisation strength is not implemented.	Redirect 302	`invalid_request`	Requested authorisation service is not implemented.
Authorisation for the given strength is implemented, but service is not available at this moment due to internal error.	Redirect 302	`server_error`	Authorisation for the given strength is implemented, but service is not available now due to internal error.
In a two-page environment, IDGW unable to assert the identity of the user.	Redirect 302	`authorisation_failure` (or) `access_denied`	User is not identified, unable to get the approval from MC User
In a two-page environment, user is identified in the first step, but rejects or cancels the approval	Redirect 302	`authorisation_failure` (or) `authorisation_denied` (or) `access_denied`	User is identified, but unable to get the approval from MC User
Binding message does not exist (or) invalid	Bad Request 400	`invalid_request`	MANDATORY parameter `binding_message` is missing
`context` parameter does not exist (or) invalid	Bad Request 400	`invalid_request`	MANDATORY parameter `context` is missing

Table 17 : MC Authorisation: Errors - Device Initiated Authorize endpoint

5.3.1.2 Error Responses: Device Initiated Token Endpoint

None. Already covered in the generic errors section.

5.3.2 Mobile Connect in Server Initiated Mode

Operator can offer Mobile Connect services in server-initiated mode where service is initiated from a Server. Error responses are returned from authorize, push notification service and resource endpoints as described below.

5.3.2.1 Error Responses: Server Initiated Authorize Endpoint

This section describes service-specific error responses in the server-initiated model.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Requested authorisation strength is not implemented.	Bad Request 400	`invalid_request`	Requested authorisation service is not implemented.
Authorisation for the given strength is implemented, but service is not available at this moment due to an internal error.	Service Unavailable 503	`server_error`	Authorisation for the given strength is implemented, but service is not available now due to an internal error.
`client_name` does not exist	Bad Request 400	`invalid_request`	MANDATORY parameter `client_name` is missing.
`client_name` exists, but value is not a registered `client_name` at Operator IDGW	Bad Request 400	`invalid_request`	Malformed request. Invalid / unregistered `client_name`.
`binding_message` does not exist (or) invalid	Bad Request 400	`invalid_request`	MANDATORY parameter `binding_message` is missing
`context` parameter does not exist (or) invalid	Bad Request 400	`invalid_request`	MANDATORY parameter `context` is missing

Table 18 : MC Authorisation: Errors - Server Initiated Authorize endpoint

5.3.2.2 Error Responses: Notification [IDGW to SP]

Error Scenario	Error code	Error Description [RECOMMENDED text]
In the single-page environment Mobile Connect User failed to approve the requested prompt (or) IDGW unable to assert the user's identity	`authorisation_failure` (or) `access_denied`	Mobile Connect user failed to approve the requested prompt.
In the single-page environment Mobile Connect User cancelled or rejected the MC authorisation request on his/her mobile device	`authorisation_denied` (or) `authorisation_failure` (or) `access_denied`	Mobile Connect user rejected / cancelled the authentication
Mobile Connect User unable to authorize, timeout occurred	`authorisation_failure` (or) `access_denied`	Timeout occurred during capturing approval from the user.
Authorisation for the given strength is implemented, but service is not available at this moment due to internal error.	`server_error`	Request authentication is temporarily not available
In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	`authorisation_failure` (or) `access_denied`	User is not identified, unable to get the approval from MC User
In a two-page environment, user is identified in the first step, but rejects or cancels the approval.	`authorisation_denied` (or) `access_denied` (or) `authorisation_failure`	User is identified, but unable to get the approval from MC User

Table 19 : MC Authorisation: Errors – Server Initiated Notification IDGW to SP

5.3.2.3 Error Responses: ACK To Notification [SP to IDGW]

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Invalid authorisation proof token [ID Token]	Bad Request 400	`invalid_request`	Mobile Connect authorisation proof token is invalid.
Invalid access token and not tied to the MC authorization proof token	Bad Request 400	`invalid_request`	Mobile Connect Authorisation access token is not valid.

Table 20 : MC Authorisation: Errors – Server Initiated ACK SP to IDGW

5.3.3 Error Responses: Resource Endpoint

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
The Access token is submitted to resource endpoint to process MC authorization transaction, and resource server finds it is invalid.	Unauthorized 401	`invalid_request` (or) `invalid_token`	Invalid token, unable to proceed with the transaction.

Table 21 : MC Authorisation: Errors - Resource Endpoint

5.4 Mobile Connect Identity Phone Number

5.4.1 Mobile Connect in Device Initiated Mode

The Operator can offer Mobile Connect Identity Phone Number service in the device-initiated mode where service is initiated through user-agent [i.e. native app, web browser, etc.,]. Error responses are returned from authorize endpoint, token and resource endpoints as described below.

5.4.1.1 Error Responses: Device Initiated Authorize Endpoint

In a two-page environment, all authentication failures must be same as described in Section 5.2.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
In the single-page environment Mobile Connect User failed to give consent (or) IDGW is unable to assert the user's identity. [authentication failure]	Redirect 302	`consent_failure (or) access_denied`	Mobile Connect user failed to give consent (or) unable to identify.
In the single-page environment Mobile Connect User cancelled or rejected the consent request on his/her consent device	Redirect 302	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect user rejected / cancelled the consent
Mobile Connect User unable to give consent timeout occurred	Redirect 302	`consent_failure (or) access_denied`	Timeout occurred during consent capture
In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	Redirect 302	`consent_failure (or) access_denied`	Mobile Connect User is not identified.
In a two-page environment, user is identified in the first step, but rejects or cancels the consent approval.	Redirect 302	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect User has denied the consent (or) consent failure.

Table 22 : MCIS Phone Number: Errors - Device Initiated Authorize Endpoint

5.4.1.2 Error Responses: Device Initiated Token Endpoint

None. Already covered in Generic errors section.

5.4.2 Mobile Connect in Server Initiated Mode

The Operator can offer Mobile Connect services in server-initiated mode where service is initiated from a Server. Error responses are returned from authorize, push notification service and resource endpoints as described below.

5.4.2.1 Error Responses: Server Initiated Authorize Endpoint

This section describes service specific error responses in the server-initiated mode.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`client_name` parameter does not exist (or) invalid	Bad Request 400	`invalid_request`	MANDATORY parameter `client_name` is missing.

Table 23 : MCIS Phone Number: Errors - Server Initiated Authorize Endpoint

5.4.2.2 Error Responses: Notification [IDGW to SP]

In a two-page environment, all server-initiated authentication failures must be same as described in Section 5.2.

Error Scenario	Error code	Error Description [RECOMMENDED text]
In the single-page environment Mobile Connect User failed to give consent (or) IDGW is unable to assert the user's identity. [authentication failure]	`consent_failure (or) access_denied`	Mobile Connect user failed to give consent (or) unable to identify.
In the single-page environment Mobile Connect User cancelled or rejected the consent request on his/her consent device	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect user rejected / cancelled the consent
Mobile Connect User unable to give consent timeout occurred	`consent_failure (or) access_denied`	Timeout occurred during consent capture for phone number.
In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	`consent_failure (or) access_denied`	Mobile Connect User is not identified.
In a two-page environment, user is identified in the first step, but rejects or cancels the consent approval.	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect User has denied the consent (or) consent failure.

Table 24 : MCIS Phone Number: Errors - Server Initiated Notification IDGW to SP

5.4.2.3 Error Responses: ACK To Notification [SP to IDGW]

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Invalid identity phone number proof token [ID Token]	Bad Request 400	`invalid_request`	Mobile Connect phone number proof token is invalid.
Invalid access token and not tied to the MC phone number proof token	Bad Request 400	`invalid_request`	Mobile Connect phone number access token is not valid.

Table 25MCIS Phone Number: Errors - Server Initiated ACK SP to IDGW

5.4.3 Error Responses: Resource Endpoint

None. Already covered in Generic errors section.

5.5 Mobile Connect Identity National ID

5.5.1 Mobile Connect in Device Initiated Mode

The Operator can offer Mobile Connect Identity NationalID service in device-initiated mode where service is initiated through user-agent [i.e. native app, web browser, etc.,]. Error responses are returned from authorize, token and resource endpoints as described below.

5.5.1.1 Error Responses: Device Initiated Authorize Endpoint

In a two-page environment, all authentication failures must be same as described in Section 5.2.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
In the single-page environment Mobile Connect User failed to give consent (or) IDGW is unable to assert the user's identity. [authentication failure] for national ID	Redirect 302	`consent_failure (or) access_denied`	Mobile Connect user failed to give consent (or) unable to identify.
In the single-page environment Mobile Connect User cancelled or rejected the consent request on his/her consent device	Redirect 302	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect user rejected / cancelled the consent for National ID
Mobile Connect User unable to give consent timeout occurred	Redirect 302	`consent_failure (or) access_denied`	Timeout occurred during consent capture
In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	Redirect 302	`consent_failure (or) access_denied`	Mobile Connect User is not identified.
In a two-page environment, user is identified in the first step, but rejects or cancels the consent approval.	Redirect 302	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect User has denied the consent (or) consent failure for MC National ID.

Table 26 : MCIS National ID: Errors – Device Initiated Authorize endpoint

5.5.1.2 Error Responses: Device Initiated Token Endpoint

None. Already covered in Generic errors section.

5.5.2 Mobile Connect in Server Initiated Mode

The Operator can offer Mobile Connect services in server-initiated mode where service is initiated from a Server. Error responses are returned from authorize endpoint, push notification service and resource endpoints as described below.

5.5.2.1 Error Responses: Server Initiated Authorize Endpoint

This section describes service-specific error responses in the server-initiated model.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`client_name` parameter does not exist	Bad Request 400	`invalid_request`	MANDATORY parameter `client_name` is missing.

Table 27 : MCIS National ID: Errors - Server Initiated Authorize endpoint

5.5.2.2 Error Responses: Notification [IDGW to SP]

In a two-page environment, all server-initiated authentication failures must be same as described in Section 5.2.

Error Scenario	Error code	Error Description [RECOMMENDED text]
In the single-page environment Mobile Connect User failed to give consent (or) IDGW is unable to assert the user's identity. [authentication failure]	`consent_failure` `(or) access_denied`	Mobile Connect user failed to give consent (or) unable to identify.
In the single-page environment Mobile Connect User cancelled or rejected the consent request on his/her consent device	`consent_denied` `(or)` `consent_failure` `(or) access_denied`	Mobile Connect user rejected / cancelled the consent
Mobile Connect User unable to give consent timeout occurred	`consent_failure` `(or) access_denied`	Timeout occurred during consent capture for National ID.
In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	`consent_failure` `(or) access_denied`	Mobile Connect User is not identified.
In a two-page environment, user is identified in the first step, but rejects or cancels the consent approval.	`consent_denied` `(or)` `consent_failure` `(or)` `access_denied`	Mobile Connect User has denied the consent (or) consent failure.

Table 28 : MCIS National ID: Errors – Server Initiated Notification IDGW to SP

5.5.2.3 Responses: ACK To Notification [SP to IDGW]

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Invalid identity NationalID proof token [JWT - ID Token]	Bad Request 400	`invalid_request`	Mobile Connect NationalId proof token is invalid.
Invalid access token and not tied to the MC NationalID proof token	Bad Request 400	`invalid_request`	Mobile Connect National ID access token is not valid.

Table 29 :MCIS National ID: Errors – Server Initiated ACK SP to IDGW

5.5.3 Error Responses: Resource Endpoint

None. Already covered in Generic errors section.

5.6 Mobile Connect Identity Sign-Up

5.6.1 Mobile Connect in Device Initiated Mode

The Operator can offer Mobile Connect Identity Sign-Up service in the device-initiated mode where service is initiated through user-agent [i.e. native app, web browser, etc.,]. Error responses are returned from authorize endpoint, token and resource endpoints as described below.

5.6.1.1 Error Responses: Device Initiated Authorize Endpoint

In a two-page environment, all authentication failures must be same as described in Section 5.2.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
In the single-page environment Mobile Connect User failed to give consent (or) IDGW is unable to assert the user's identity. [authentication failure]	Redirect 302	`consent_failure (or) access_denied`	Mobile Connect user failed to give consent (or) unable to identify.
In the single-page environment Mobile Connect User cancelled or rejected the consent request on his/her consent device	Redirect 302	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect user rejected / cancelled the consent
Mobile Connect User unable to give consent timeout occurred	Redirect 302	`consent_failure (or) access_denied`	Timeout occurred during consent capture
In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	Redirect 302	`consent_failure (or) access_denied`	Mobile Connect User is not identified.
In a two-page environment, user is identified in the first step, but rejects or cancels the consent approval.	Redirect 302	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect User has denied the consent (or) consent failure.

Table 30 : MCIS Sign-Up: Errors - Device Initiated Authorize Endpoint

5.6.1.2 Error Responses: Device Initiated Token Endpoint

None. Already covered in Generic errors section.

5.6.2 Mobile Connect in Server Initiated Mode

The Operator can offer Mobile Connect services in server-initiated mode where service is initiated from a Server. Error responses are returned from authorize endpoint, push notification service and resource endpoints as described below.

5.6.2.1 Error Responses: Server Initiated Authorize Endpoint

This section describes service-specific error responses in the server-initiated model.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`client_name` parameter does not exist (or) invalid	Bad Request 400	`invalid_request`	MANDATORY parameter `client_name` is missing.

Table 31 : MCIS Signup: Errors – Server Initiated Authorize Endpoint

5.6.2.2 Error Responses: Notification [IDGW to SP]

In a two-page environment, all server-initiated authentication failures must be same as described in Section 5.2.

Error Scenario	Error code	Error Description [RECOMMENDED text]
In the single-page environment Mobile Connect User failed to give consent (or) IDGW is unable to assert the user's identity. [authentication failure]	`consent_failure (or) access_denied`	Mobile Connect user failed to give consent (or) unable to identify.
In the single-page environment Mobile Connect User cancelled or rejected the consent request on his/her consent device	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect user rejected / cancelled the consent
Mobile Connect User unable to give consent timeout occurred	`consent_failure (or) access_denied`	Timeout occurred during consent capture
In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	`consent_failure (or) access_denied`	Mobile Connect User is not identified.
In a two-page environment, user is identified in the first step, but rejects or cancels the consent approval.	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect User has denied the consent (or) consent failure.

Table 32 : MCIS Sign-Up: Errors – Server Initiated Notification IDGW to SP

5.6.2.3 Error Responses: ACK To Notification [SP to IDGW]

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Invalid identity Signup proof token [JWT - ID Token]	Bad Request 400	`invalid_request`	Mobile Connect Signup proof token is invalid.
Invalid access token and not tied to the MC Signup proof token	Bad Request 400	`invalid_request`	Mobile Connect Signup access token is not valid.

Table 33 : MCIS Sign-Up: Errors - Server Initiated ACK SP to IDGW

5.6.3 Error Responses: Resource Endpoint

None. Already covered in Generic errors section.

5.7 Mobile Connect Attributes Know Your Customer [KYC]

5.7.1 Mobile Connect in Device Initiated Mode

The Operator can offer Mobile Connect Attributes KYC service in device-initiated mode where service is initiated through user-agent [i.e. native app, web browser etc.,]. Error responses are returned from authorize endpoint, token and resource endpoints as described below.

5.7.1.1 Error Responses: Device Initiated Authorize Endpoint

In a two-page environment, all authentication failures must be same as described in Section 5.2.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`claims` parameter exists, but MANDATORY parameters inside claims are missing (or) `claims` parameter exist, but the value is empty (or) `claims` parameter does not exist	Redirect 302	`invalid_request`	MANDATORY values in the claims parameter are missing for MC KYC service (or) invalid
`client_name` parameter does not exist (or) invalid	Redirect 302	`invalid_request`	MANDATORY parameter `client_name` is missing.
In the single-page environment Mobile Connect User failed to give consent (or) IDGW is unable to assert the user's identity. [authentication failure]	Redirect 302	`consent_failure (or) access_denied`	Mobile Connect user failed to give consent (or) unable to identify.
In the single-page environment Mobile Connect User cancelled or rejected the consent request on his/her consent device	Redirect 302	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect user rejected / cancelled the consent
Mobile Connect User unable to give consent timeout occurred	Redirect 302	`consent_failure (or) access_denied`	Timeout occurred during consent capture
In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	Redirect 302	`consent_failure (or) access_denied`	Mobile Connect User is not identified.
In a two-page environment, user is identified in the first step, but rejects or cancels the consent approval.	Redirect 302	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect User has denied the consent (or) consent failure.

Table 34 : MC KYC: Errors – Device Initiated Authorize Endpoint

5.7.1.2 Error Responses: Device Initiated Token Endpoint

None. Already covered in Generic errors section.

5.7.2 Mobile Connect in Server Initiated Mode

The Operator can offer Mobile Connect services in server-initiated mode where service is initiated from a Server. Error responses are returned from authorize endpoint, push notification service and resource endpoints as described below.

5.7.2.1 Error Responses: Server Initiated Authorize Endpoint

This section describes service-specific error responses in the server-initiated model.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`client_name` parameter does not exist (or) invalid	Bad Request 400	`invalid_request`	MANDATORY parameter `client_name` is missing.
`claims` parameter does not exist (or) `claims` parameter exist but MANDATORY parameters inside claims are missing (or) `claims` parameter exist but the value is empty	Bad Request 400	`invalid_request`	MANDATORY parameter claims are missing in the request for MC KYC service (or) invalid

Table 35 : MC KYC: Errors – Server Initiated Authorize Endpoint

5.7.2.2 Error Responses: Notification [Operator IDGW to SP]

In a two-page environment, all server-initiated authentication failures must be same as described in Section 5.2.

Error Scenario	Error code	Error Description [REOMMENDED text]

In the single-page environment Mobile Connect User failed to give consent (or) IDGW is unable to assert the user's identity. [authentication failure]	`consent_failure`
	`(or)`
	`access_denied`
		Mobile Connect user failed to give consent (or) unable to identify.

In the single-page environment Mobile Connect User cancelled or rejected the consent request on his/her consent device	`consent_denied`
	`(or)`
	`consent_failure`
	`(or)`
	`access_denied`
		Mobile Connect user rejected / cancelled the consent

Mobile Connect User unable to give consent timeout occurred	`consent_failure`
	`(or)`
	`access_denied`
		Timeout occurred during consent capture

In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	`consent_failure`
	`(or)`
	`access_denied`
		Mobile Connect User is not identified.

In a two-page environment, user is identified in the first step, but rejects or cancels the consent approval.	`consent_denied`
	`(or)`
	`consent_failure`
	`(or)`
	`access_denied`
		Mobile Connect User has denied the consent (or) consent failure.

Table 36 : MC KYC: Errors - Server Initiated Notification IDGW to SP

5.7.2.3 Error Responses: ACK To Notification [SP to IDGW]

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Invalid identity KYC proof token [JWT - ID Token]	Bad Request 400	`invalid_request`	Mobile Connect KYC proof token is invalid.
Invalid access token and not tied to the MC KYC proof token	Bad Request 400	`invalid_request`	Mobile Connect KYC access token is not valid.

Table 37 : MC KYC: Errors - Server Initiated ACK SP to IDGW

5.7.3 Error Responses: Resource Endpoint

None. Already covered in Generic errors section.

5.8 Mobile Connect Attributes Account Takeover Protection [ATP]

MC ATP MUST operate in server-initiated mode. By default, no prompt will be displayed to the user. If IDGW policy mandates to capture the consent from MC User, then it will be prompted.

5.8.1 Mobile Connect in Device Initiated Mode

MC ATP product does not operate in the device-initiated mode.

5.8.2 Mobile Connect in Server Initiated Mode

The Operator can offer Mobile Connect services in server-initiated mode where service is initiated from a Server. Error responses are returned from authorize endpoint, push notification service and resource endpoints as described below.

5.8.2.1 Error Responses: Server Initiated Authorize Endpoint

This section describes service-specific error responses in the server-initiated model.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`client_name` parameter does not exist (or) invalid	Bad Request 400	`invalid_request`	MANDATORY parameter `client_name` is missing.
`claims` parameter does not exist (or) `claims` parameter exist but MANDATORY parameters inside claims are missing (or) `claims` parameter exist but value is empty	Bad Request 400	`invalid_request`	MANDATORY parameter claims are missing for MC ATP service (or) invalid

Table 38 : MC ATP: Errors – Server Initiated Authorize Endpoint

5.8.2.2 Error Responses: Notification [Operator IDGW to SP]

In a two-page environment, all server-initiated authentication failures must be same as described in Section 5.2.

Error Scenario	Error code	Error Description [RECOMMENDED text]
In the single-page environment Mobile Connect User failed to give consent (or) IDGW is unable to assert the user's identity. [authentication failure]	`consent_failure` (or) `access_denied`	Mobile Connect user failed to give consent (or) unable to identify.
In the single-page environment Mobile Connect User cancelled or rejected the consent request on his/her consent device	`consent_denied` (or) `consent_failure` (or) `access_denied`	Mobile Connect user rejected / cancelled the consent
Mobile Connect User unable to give consent timeout occurred	`consent_failure` (or) `access_denied`	Timeout occurred during consent capture
In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	`consent_failure` (or) `access_denied`	Mobile Connect User is not identified.
In a two-page environment, user is identified in the first step, but rejects or cancels the consent approval.	`consent_denied` (or) `consent_failure` (or) `access_denied`	Mobile Connect User has denied the consent (or) consent failure.

Table 39 : MC ATP: Errors - Server Initiated Notification IDGW to SP

5.8.2.3 Error Responses: ACK To Notification [SP to IDGW]

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
Invalid identity ATP proof token [JWT - ID Token]	Bad Request 400	`invalid_request`	Mobile Connect ATP proof token is invalid.
Invalid access token and not tied to the MC ATP proof token	Bad Request 400	`invalid_request`	Mobile Connect ATP access token is not valid.

Table 40 : MC ATP: Errors - Server Initiated ACK SP to IDGW

5.8.3 Error Responses: Resource Endpoint

None. Already covered in Generic errors section.

5.9 Mobile Connect Attributes Verified MSISDN Match & Share

5.9.1 Mobile Connect in Device Initiated Mode

The Operator can offer Mobile Connect Attributes VM service in device-initiated mode only where service is initiated through user-agent [i.e. native app, web browser, etc.,]. Error responses are returned from authorize endpoint, token and resource endpoints as described below.

5.9.1.1 Error Responses: Device Initiated Authorize Endpoint

The following errors are applicable if Operator captures the consent.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
In the single-page environment Mobile Connect User failed to give consent (or) IDGW is unable to assert the user's identity. [authentication failure]	Redirect 302	`consent_failure (or) access_denied`	Mobile Connect user failed to give consent (or) unable to identify.
In the single-page environment Mobile Connect User cancelled or rejected the consent request on his/her consent device	Redirect 302	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect user rejected / cancelled the consent
Mobile Connect User unable to give consent timeout occurred	Redirect 302	`consent_failure (or) access_denied`	Timeout occurred during consent capture
In a two-page environment, IDGW failed to assert the user's identity the first step when prompted for authentication.	Redirect 302	`consent_failure (or) access_denied`	Mobile Connect User is not identified.
In a two-page environment, user is identified in the first step, but rejects or cancels the consent approval.	Redirect 302	`consent_denied (or) consent_failure (or) access_denied`	Mobile Connect User has denied the consent (or) consent failure.
Network MSISDN is not available in the HTTP header	Redirect 302	`access_denied`	Device MSISDN is not available.

Table 41 : MC VM: Errors – Device Initiated Authorize Endpoint

5.9.1.2 Error Responses: Device Initiated Token Endpoint

None. Already covered in Generic Errors section.

5.9.2 Error Responses: Resource Endpoint

Resource endpoint common errors are already covered in Generic section.

Error Scenario	HTTP mode	Error code	Error Description [RECOMMENDED text]
`mc_claims` parameter does not exist	Bad Request 400	`invalid_request`	MANDATORY parameter `mc_claims` are missing
`mc_claims` parameter exist but has MANDATORY parameter device_msisdn is missing [VM share]	Bad Request 400	`invalid_request`	MANDATORY parameter is missing from `mc_claims`.
`mc_claims` parameter exist, but has no entries.	Bad Request 400	`invalid_request`	MANDATORY parameter values from `mc_claims` are missing.

Table 42 : MC VM: Errors – Resource Endpoint

6 MCIS & Attribute Services Names list

This section provides information about normative claim names for the Mobile Connect Identity Services.

6.1 Mobile Connect Phone Number

This section provides MCIS Phone number product claim names that must be used in the MCIS Phone number product OIDC request and responses.

Attribute Name	Description
phone_number	User's Mobile Connect designated mobile number

Table 43 : MCIS Phone Number Attributes

6.2 Mobile Connect Identity Services Sign-up

This section provides MCIS Sign-up product claim names that must be used in the MCIS Signup product OIDC request and responses.

Attribute Name	MANDATORY	Description
phone_number_alternate	Yes	User's alternate/secondary telephone number [E.164]
title	No	Salutation
given_name	Yes	Given name(s) or first names
family_name	Yes	Surname(s) or last name(s) of the user
middle_name	No	Middle name(s) of the user
street_address	Yes	User's street (incl. house name/number)
city	No	User's city
state	No	User's State / County
postal_code	Yes	User's Zip/ Postcode
country	Yes	User's postal country
email	No	User's e-mail

Table 44 : MCIS Sign Up Attributes

6.3 Mobile Connect Identity Services National ID

This section provides MCIS National ID product claim names that must be used in the MCIS National ID product OIDC request and responses.

Attribute Name	MANDATORY	Description
phone_number	No	User's Mobile Connect designated mobile number.
title	No	Salutation
given_name	Yes	Given name(s) or first names
family_name	Yes	Surname(s) or last name(s) of the user
middle_name	No	Middle name(s) of the user

Attribute Name	MANDATORY	Description
street_address	Yes	User's street (incl. house name/number)
state	No	User's State / County
postal_code	No	User's Zip/ Postcode
country	No	User's postal country
email	No	User's e-mail
birthdate	Yes	User's birth date
national_identifier	Yes	User's Identifier (eIDAS), any national identifier like Social Security Identifier, passport e t c. (depends on the local regulations)

Table 45 : MCIS National ID Attributes

6.4 MC KYC Attribute List

This section provides MC KYC (Know Your Customer) Attribute list

Response Match Values
"Y"– match is successful "N-NA"- match failed, data is not available "N-AV"– match failed; data is available "N-AD"– match failed, data is available but access is denied

Table 46 : MC KYC Response Values

The following table identifies the request attributes for the plain text KYC Match service variant; for the hashed variant, the attribute names SHOULD be appended with "_hash"(i.e., given_name_hash).

Attribute Name	Usage	Description
given_name	**MANDATORY** ([name] (OR) [given_name, family_name])	Given name(s) or first name(s) of the End-User. Note that in some cultures, people can have multiple given names; all can be present, with the names being separated by space characters. Always used in conjunction with the family_name attribute.
family_name		Family name(s), surname(s) or last name(s) of the End-User. Note that in some cultures, people can have multiple family names or no family name; all can be present, with the names being separated by space characters. Always used in conjunction with the given_name attribute.
name		concatenated given_name and family_name.
address		concatenated houseno_or_housename,

Attribute Name	Usage	Description
	MANDATORY ([address] (OR) [houseno_or_housename, postal_code, town, country]	postal_code and optionally town and country.
houseno_or_housename		Registered house number or house name
postal_code		Registered Zip code or post code
town		Registered city or town name
country		Registered country
birthdate	OPTIONAL	End-User's birthday, represented as an **ISO 8601:2004** [ISO8601-2004] YYYY-MM-DD format. The year MAY be 0000, indicating that it is omitted.

Table 47 MCKYC Request Attributes

For each attribute, if there is a successful match, the attribute name+value SHOULD be echoed back to the TSP in the response.

Attribute Name	Usage
given_name	**MANDATORY** ([name] (OR) [given_name, family_name])
family_name	
name	
address	**MANDATORY** ([address] (OR) [houseno_or_housename, postal_code, town, country]
houseno_or_housename	
postal_code	
town	
country	
birthdate	OPTIONAL
given_name_match	**MANDATORY** ([name_match] (OR) [given_name_match, family_name_match])
family_name_match	
name_match	
address_match	**MANDATORY** ([address_match] (OR) [houseno_or_housename_match, postal_code_match, town_match, country_match]]
houseno_or_housename_match	
postal_code_match	
town_match	
country_match	
birthdate_match	OPTIONAL

Attribute Name	Usage	Response values
`is_lost_stolen`	OPTIONAL	Allowed boolean values are `true / false`
`billing_segment`	OPTIONAL	Allowed values `"PAYG","PAYM", "Business"`
`account_state`	OPTIONAL	Allowed values `"active"/ "inactive"`

Table 48 : MC KYC Response attributes

6.5 MC ATP Attribute List

The attribute set returned within the Account Takeover Protection service is described in this section. The ID GW SHOULD only offer the ATP service if it has access to all the MANDATORY attribute values.

Attribute Name	Usage Category	Description
`is_unconditional_call_divert_active`	MANDATORY	Mobile phone account has an unconditional call divert set to a number; allowed boolean values true or false
`is_lost_stolen`	OPTIONAL	Boolean values true or false
`sim_change`	MANDATORY	Timestamp[1] of last MSISDN <-> IMSI pairing change
`device_change`	OPTIONAL	Timestamp[2] of last MSISDN <-> IMEI pairing change
`account_state`	OPTIONAL	"active"or "inactive"[3].

Table 49 : MC ATP Attribute Set

6.6 MC VM Attribute List

This section describes the attributes used in MC VM Share and MC VM Match services.

6.6.1 Mobile Connect Verified MSISDN Share

For Mobile Connect Verified MSISDN Share any one of the following parameters must be supported based on the service variant being requested (see section **Error! Reference source not found.**). The values must be returned through attribute share endpoint.

[1] It must adhere to the RFC 3339 absolute timestamp format.

[2] Please refer above footnote [2].

[3] The values always depend on mobile account status.

Attribute name	Type	Description
device_msisdn	string	MSISDN returned to the SP (via UserInfo). **E.164** [E.164] is RECOMMENDED as the format of this Claim, for example, +441234567890

Table 50 : MC VM Attribute Set

6.6.2 Mobile Connect Verified MSISDN Match

6.6.2.1 Resource Request Using 'claims' Parameters

For Mobile Connect Verified MSISDN Match any one of the following claims must exist in the resource request.

Claim parameter name	Type	Description
device_msisdn	String	The value is the MSISDN to verify. [E.164] is RECOMMENDED as the format of this Claim, for example, +441234567890
device_msisdn_hash	String	Hashed value of device_msisdn. SHA256 algorithm is RECOMMENDED for backward interoperability to current implementations[4]. New implementations SHOULD NOT use SHA256[5] unless backward interoperability is required. Hashing algorithms can be negotiated offline[6]

Table 51: MC VM Resource Request Attribute Set

6.6.2.2 Resource Response Parameters

Claim parameter name	Type	Usage Category	Description

[4] Changing RECOMMENDED hashing algorithm for interoperability is for future study by CPAS.

[5] SHA256 is fast but to mitigate brute force attacks on the hash the hashing algorithms should be slow like e.g. PBKDF2

[6] Currently identified algorithms are PBKDF2, SHA256_crypt, Argon2 and SHA256. An operator and SP can negotiate any of these algorithms offline. Changing RECOMMENDED algorithms for interoperability is for future study.

device_msisdn_verified	Boolean	MANDATORY	True or False. Match result.

Table 52: MC VM Resource Endpoint Response

7 Mobile Connect Provider Metadata

This section describes required Mobile Connect Provider Metadata configuration based on OpenID Provider Metadata and with Mobile Connect specific amendments. The RECOMMENDED location to host provider metadata is:

https://idgw-operator.example.com/mc_examplepath/.well-known/openid-configuration

7.1.1 OpenID Provider Metadata

Field Name	Usage Category	Description
issuer	MANDATORY	Same as defined in [5]
authorization_endpoint	MANDATORY	The device-initiated authorization endpoint only. It is same as defined in [5].
bc_authorize_endpoint	MANDATORY	The server-initiated authorization end point only. It is same as defined in [xx].
token_endpoint	MANDATORY	Same as defined in [5]
jwks_uri	MANDATORY	Same as defined in [5]
scopes_supported	MANDATORY	The default scope "openid" must be supported. This parameter must contain all the Mobile Connect scopes supported by the Operator IDGW. Operators must register their own Mobile Connect innovation product scopes before exposing them through this parameter. The syntax is same as defined in the [5].
response_types_supported	MANDATORY	This parameter must list down all response types used in Mobile Connect. [i.e. for FC and BC]. The syntax is same as defined in [5].
grant_types_supported	MANDATORY	This parameter must list down all Mobile Connect grant_types. The syntax is same as defined in [5]. It must have a value "authorization_code"
acr_values_supported	MANDATORY	It must have Mobile Connect acr values i.e. [1,2,3,4] based on ISO29115. The syntax is same as defined in [5].
id_token_signing_alg_values_supported	MANDATORY	Same as defined in [5]
id_token_encryption_alg_values_supported	OPTIONAL [for future]	Same as defined in [5]. Placeholder for future requirements.
id_token_encryption_enc_values_supported	OPTIONAL [for future]	Same as defined in [5]. Placeholder for future requirements.
request_object_signing_alg_values_supported	OPTIONAL [MANDATORY if server-initiated is supported]	Same as defined in [5].
request_object_enryption_alg_values_supported	OPTIONAL	Same as defined in [5]. Place holder for future requirements.
claims_parameter_supported	OPTIONAL [MANDATORY if MC KYC and ATP are supported]	Same as defined in [5]. This parameter must be true, if MC KYC and MC ATP products are supported.

Field Name	Usage Category	Description
request_parameter_supported	OPTIONAL [MANDATORY if server-initiated is supported]	Same as defined in [5].
ui_locales_supported	MANDATORY	Same as defined in [5].

Table 53 : MC Provider Metadata [Derived from OIDC]

7.2 Hashing Algorithms

This section lists hashing algorithms that can be used to obfuscate attribute values, allowing an exact match to be recognised without directly revealing the value. The identifiers can appear in the mc_hash_algs_supported metadata member.

The only currently defined value is "SHA-256". It is used by early implementations but SHOULD NOT be used unless backwards compatibility with those implementations is required as the hash algorithm in this situation SHOULD be deliberately slow to mitigate dictionary attacks. Better algorithms exist (such as PBKDF2 and Argon2id). It is expected that a better algorithm will be RECOMMENDED in the future.

Hashing Algorithm	Identifier
SHA-256, as hex digits	SHA-256
PBKDF2 using HMAC with SHA-256	PBKDF2-HMAC-SHA256[7]
Argon2id	ARGON2ID[8]

Table 54 : RECOMMENDED Hashing Algorithms

7.3 Mobile Connect Specific Provider Metadata Parameters

This section describes Mobile Connect specific parameters.

Field Name	Usage Category	Description
mc_version	MANDATORY	A JSON array containing the list of the Mobile Connect profile versions [i.e. mc_v1.0, mc_v1.1, mc_v2.0 etc]
mc_amr_values_supported	MANDATORY	The parameter must all amr values supported by the Operator IDGW
mc_hash_algs_supported	MANDATORY	JSON array containing a list of hash algorithms supported by the MC provider, for where there a requirement in MC services to obfuscate attribute values [e.g., MC KYC, MC VM etc.,]. See section 7.2.
mc_di_scopes_supported	MANDATORY	This parameter will list all MC services supported by the IDGW in device-initiated mode

[7] CPAS defined identifier, once PBKDF2 standard sets out a new one, MC specs will adopt the same. Implementation details are out of scope for this document.

[8] CPAS defined identifier, once ARGON2ID standard sets out a new one, MC specs will adopt the same. Implementation details are out of scope for this document.

Field Name	Usage Category	Description
mc_si_scopes_supported	MANDATORY	This parameter will list all MC services supported by the IDGW in server-initiated mode.
mc_claims_paramter_supported	MANDATORY	'true'; if mc_claims is supported in the resource call 'false': if mc_claims is NOT supported in the resource call. The mc_claims parameter is used to request that specific claims be returned from resource end point. The value is a JSON object listing the requested Claims and is part of the resource request.
login_hint_types_supported	MANDATORY	The values must be MSISDN, ENCR_MSISDN and PCR

Table 55 : MC Specific Provider Metadata Parameters

The following is an example of Mobile Connect Provider Metadata configuration.

```
HTTP/1.1 200 OK
Content-Type: application/json
  {
        "issuer": "https://mc-idgw-operator.example.com",
        "authorization_endpoint":
                "https://mc-idgw-operator.example.com/connect/authorize",
        "token_endpoint":
                "https://mc-idgw-operator.example.com/connect/token",
        "premiuminfo_endpoint":
                "https://server.example.com/connect/userinfo",
        "jwks_uri": "https://mc-idgw-operator.example.com/jwks.json",
        "scopes_supported": ["openid","mc_authn","mc_authz","mc_kyc_hash"],
        "response_types_supported": ["code", "mc_bc_async_code"],
        "acr_values_supported": ["2", "3"],
        "id_token_signing_alg_values_supported": ["RS256","ES256",
"HS256"],
        "request_object_signing_alg_values_supported": ["none","RS256",
"ES256"],
        "claims_parameter_supported": true,
        "ui_locales_supported": ["en-US","en-GB","en-CA","fr-FR", "fr-CA"],
        "mc_version": ["mc_v1.0", "mc_v1.1", "mc_v2.0", "mc_v2 .2"],
        "mc_amr_values_suppored": ["SIM_OK", "SIM_PIN", "FIDO_OK",
"FIDO_PIN"],
        "mc_hash_algs_supported": ["SHA-256"],
        "mc_di_scope_values_supported": ["openid mc_authn", "openid
mc_attr_vm_share"],
        "mc_si_scope_values_supported": ["openid mc_authn", "openid
mc_atp"],
        "mc_claims_parameter_supported": true
  }
```

Annex A Document Management

A.1 Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
1.0	27/05/2016	New document	PDATA/PSMC	Siva (Venkatasivakumar Boyalakuntla) / GSMA
1.1	12/05/2017	Transfer of PRD from Personal Data		Nick Cheung / GSMA
2.0	11/08/2017	Major update with new error messages (member´s feedback) and reorganised per Mobile Connect product	TG	Venkatasivakumar Boyalakuntla / GSMA

A.2 Other Information

Type	Description
Document Owner	IDG
Editor / Company	Yolanda Sanz/GSMA

It is our intention to provide a quality product for your use. If you find any errors or omissions, please contact us with your comments. You may notify us at prd@gsma.com

Your comments or suggestions & questions are always welcome.