Login Service Interface Control Document

EOEPCA.ICD.xxx

TVUK System Team

Version 1.0, 30/11/2020:

Login Service Interface Control Document

EO Exploitation Platform Common Architecture

Login Service Interface Control Document

EOEPCA.ICD.xxx

COMMENTS and ISSUES	**PDF**
If you would like to raise comments or issues on this document, please do so by raising an Issue at the following URL https://github.com/EOEPCA/um-login-service/issues.	This document is available in PDF format here.
EUROPEAN SPACE AGENCY CONTRACT REPORT The work described in this report was done under ESA contract. Responsibility for the contents resides in the author or organisation that prepared it.	**TELESPAZIO VEGA UK Ltd** 350 Capability Green, Luton, Bedfordshire, LU1 3LU, United Kingdom. Tel: +44 (0)1582 399000 www.telespazio-vega.com

AMENDMENT HISTORY

This document shall be amended by releasing a new edition of the document in its entirety. The Amendment Record Sheet below records the history and issue status of this document.

Table 1. Amendment Record Sheet

ISSUE	DATE	REASON
0.1	dd/mm/yyyy	Initial in-progress draft

Chapter 1. Introduction

1.1. Purpose and Scope

This document presents the Login Service Interfaces for the Common Architecture. It servers as a complementary document to its corresponding Software Design Document.

Chapter 2. Overview

This Interface Control Document (ICD) is a companion to the System Design Document for the Login Service [UM-LOGIN-SERVICE-SDD]. The ICD provides a Building Block level specification of the interfaces exposed by the Login Service to the rest of EOEPCA components.

Section Login Service Interfaces

Provides the interface specification of the Building Block.

Chapter 3. Login Service Interfaces

Abstract

OpenID Connect Provider (OP) & UMA Authorization Server (AS)

3.1. Endpoints

3.1.1. OIDC - Authentication

3.1.1.1. endSession

`GET /end_session`

End current session.

3.1.1.1.1. Description

End current session.

3.1.1.1.2. Parameters

Query Parametersa

Name	Description	Required
id_token_hint	Previously issued ID Token (id_token) passed to the logout endpoint as a hint about the End-User's current authenticated session with the Client. This is used as an indication of the identity of the End-User that the RP is requesting be logged out by the OP. The OP need not be listed as an audience of the ID Token when it is used as an id_token_hint value.	-
post_logout_redirect_uri	URL to which the RP is requesting that the End-User's User Agent be redirected after a logout has been performed. The value MUST have been previously registered with the OP, either using the post_logout_redirect_uris Registration parameter or via another mechanism. If supplied, the OP SHOULD honor this request following the logout.	-

Name	Description	Required
state	Opaque value used by the RP to maintain state between the logout request and the callback to the endpoint specified by the post_logout_redirect_uri parameter. If included in the logout request, the OP passes this value back to the RP using the state query parameter when redirecting the User Agent back to the RP.	-
session_id	Session Id	-

3.1.1.1.3. Return Type

-

3.1.1.1.4. Content Type

- application/json

3.1.1.1.5. Responses

Table 2. http response codes

Code	Message	Datatype
200	OK - User redirected to logout page	<<>>
302	Resource Found.	<<>>
400	Error codes for end session endpoint.	*EndSessionError*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.1.1.6. Samples

3.1.1.2. GET Authorize

`GET /authorize`

The Authorization Endpoint performs Authentication of the End-User.

3.1.1.2.1. Description

End-User Authentication and Authorization done by sending the User Agent to the Authorization Endpoint using request parameters defined by OAuth 2.0 and OpenID Connect.

3.1.1.2.2. Parameters

Query Parameters

Name	Description	Required
scope	OpenID Connect requests MUST contain the openid scope value. If the openid scope value is not present, the behavior is entirely unspecified. Other scope values MAY be present.	X
response_type	OAuth 2.0 Response Type value that determines the authorization processing flow to be used, including what parameters are returned from the endpoints used.	X
client_id	OAuth 2.0 Client Identifier valid at the Authorization Server.	X
redirect_uri	Redirection URI to which the response will be sent. This URI MUST exactly match one of the Redirection URI values for the Client pre-registered at the OpenID Provider.	X
state	Opaque value used to maintain state between the request and the callback.	-
response_mode	Informs the Authorization Server of the mechanism to be used for returning parameters from the Authorization Endpoint.	-
nonce	String value used to associate a Client session with an ID Token, and to mitigate replay attacks.	-
display	ASCII string value that specifies how the Authorization Server displays the authentication and consent user interface pages to the End-User.	-
prompt	Space delimited, case sensitive list of ASCII string values that specifies whether the Authorization Server prompts the End-User for reauthentication and consent. The defined values are - none, login, consent, select_account.	-
max_age	Maximum Authentication Age. Specifies the allowable elapsed time in seconds since the last time the End-User was actively authenticated by the OP.	-
ui_locales	End-User's preferred languages and scripts for the user interface, represented as a space-separated list of BCP47 [RFC5646] language tag values, ordered by preference.	-

Name	Description	Required
id_token_hint	ID Token previously issued by the Authorization Server being passed as a hint about the End-User's current or past authenticated session with the Client. If the End-User identified by the ID Token is logged in or is logged in by the request, then the Authorization Server returns a positive response.	-
login_hint	Hint to the Authorization Server about the login identifier the End-User might use to log in (if necessary).	-
acr_values	Requested Authentication Context Class Reference values. Space-separated string that specifies the acr values that the Authorization Server is being requested to use for processing this Authentication Request, with the values appearing in order of preference.	-
amr_values	AMR Values.	-
request	This parameter enables OpenID Connect requests to be passed in a single, self-contained parameter and to be optionally signed and/or encrypted. The parameter value is a Request Object value. It represents the request as a JWT whose Claims are the request parameters.	-
request_uri	This parameter enables OpenID Connect requests to be passed by reference, rather than by value. The request_uri value is a URL using the https scheme referencing a resource containing a Request Object value, which is a JWT containing the request parameters.	-
request_session_id	Request session id.	-
session_id	Session id of this call.	-
origin_headers	Origin headers. Used in custom workflows.	-
code_challenge	PKCE code challenge.	-
code_challenge_method	PKCE code challenge method.	-
custom_response_headers	Custom Response Headers.	-
claims	Requested Claims.	-
auth_req_id	CIBA authentication request Id.	-

3.1.1.2.3. Return Type

-

3.1.1.2.4. Content Type

- application/json

3.1.1.2.5. Responses

Table 3. http response codes

Code	Message	Datatype
200	OK	<<>>
302	Error codes for authorization endpoint.	*AuthorizeError*
400	Invalid parameters are provided to endpoint.	*ErrorResponse*
401	Unauthorized access request.	*ErrorResponse*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.1.2.6. Samples

3.1.1.3. GET Clientinfo

`GET /clientinfo`

To get Claims details about the registered client.

3.1.1.3.1. Description

The ClientInfo Endpoint is an OAuth 2.0 Protected Resource that returns Claims about the registered client.

3.1.1.3.2. Parameters

Header Parameters

Name	Description	Required
Authorization		-

Query Parameters

Name	Description	Required
access_token		-

3.1.1.3.3. Return Type

ClientInfoResponse

3.1.1.3.4. Content Type

- application/json

3.1.1.3.5. Responses

Table 4. http response codes

Code	Message	Datatype
200	OK	*ClientInfoResponse*
400	Invalid Request are provided to endpoint.	[inline_response_400]

3.1.1.3.6. Samples

3.1.1.4. GET Introspection

`GET /introspection`

The Introspection OAuth 2 Endpoint.

3.1.1.4.1. Description

The Introspection OAuth 2 Endpoint.

3.1.1.4.2. Parameters

Header Parameters

Name	Description	Required
Authorization	Client Authorization details that contains the access token along with other details.	X

Query Parameters

Name	Description	Required
token		X
token_type_hint	ID Token previously issued by the Authorization Server being passed as a hint about the End-User.	-
response_as_jwt	OPTIONAL. Boolean value with default value false. If true, returns introspection response as JWT (signed based on client configuration used for authentication to Introspection Endpoint).	-

3.1.1.4.3. Return Type

IntrospectionResponse

3.1.1.4.4. Content Type

- application/json

3.1.1.4.5. Responses

Table 5. http response codes

Code	Message	Datatype
200	OK	*IntrospectionResponse*
400	Error codes for introspection endpoint.	*AuthorizeError*
401	Unauthorized access request.	*ErrorResponse*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.1.4.6. Samples

3.1.1.5. GET Userinfo

`GET /userinfo`

Returns Claims about the authenticated End-User.

3.1.1.5.1. Description

Returns Claims about the authenticated End-User.

3.1.1.5.2. Parameters

Header Parameters

Name	Description	Required
Authorization		-

Query Parameters

Name	Description	Required
access_token	OAuth 2.0 Access Token.	X

3.1.1.5.3. Return Type

[Object]

3.1.1.5.4. Content Type

- application/jwt
- application/json

3.1.1.5.5. Responses

Table 6. http response codes

Code	Message	Datatype
200	OK	[Object]
400	Invalid parameters provided to endpoint.	[inline_response_400_6]
401	Invalid parameters provided to endpoint.	[inline_response_401]
403	Invalid parameters provided to endpoint.	[inline_response_403]
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.1.5.6. Samples

3.1.1.6. jwks

`GET /jwks`

A JSON Web Key (JWK) used by server. JWK is a JSON data structure that represents a set of public keys as a JSON object [RFC4627].

3.1.1.6.1. Description

Provides list of JWK used by server.

3.1.1.6.2. Parameters

3.1.1.6.3. Return Type

WebKeysConfiguration

3.1.1.6.4. Content Type

- application/json

3.1.1.6.5. Responses

Table 7. http response codes

Code	Message	Datatype
200	OK	*WebKeysConfiguration*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.1.6.6. Samples

3.1.1.7. POST Authorize

`POST /authorize`

The Authorization Endpoint performs Authentication of the End-User.

3.1.1.7.1. Description

End-User Authentication and Authorization done by sending the User Agent to the Authorization Endpoint using request parameters defined by OAuth 2.0 and OpenID Connect.

3.1.1.7.2. Parameters

3.1.1.7.3. Form Parameter

Name	Description	Required
scope	OpenID Connect requests MUST contain the openid scope value. If the openid scope value is not present, the behavior is entirely unspecified. Other scope values MAY be present. [string]	X
response_type	OAuth 2.0 Response Type value that determines the authorization processing flow to be used, including what parameters are returned from the endpoints used. [string]	X
client_id	OAuth 2.0 Client Identifier valid at the Authorization Server. [string]	X
redirect_uri	Redirection URI to which the response will be sent. This URI MUST exactly match one of the Redirection URI values for the Client pre-registered at the OpenID Provider. [string]	X
state	Opaque value used to maintain state between the request and the callback. [string]	-
response_mode	Informs the Authorization Server of the mechanism to be used for returning parameters from the Authorization Endpoint. [string]	-
nonce	String value used to associate a Client session with an ID Token, and to mitigate replay attacks. [string]	-
display	ASCII string value that specifies how the Authorization Server displays the authentication and consent user interface pages to the End-User. [string]	-
prompt	Space delimited, case sensitive list of ASCII string values that specifies whether the Authorization Server prompts the End-User for reauthentication and consent. [string]	-

Name	Description	Required
max_age	Maximum Authentication Age. Specifies the allowable elapsed time in seconds since the last time the End-User was actively authenticated by the OP. [integer]	-
ui_locales	End-User's preferred languages and scripts for the user interface, represented as a space-separated list of BCP47 [RFC5646] language tag values, ordered by preference. [string]	-
id_token_hint	ID Token previously issued by the Authorization Server being passed as a hint about the End-User's current or past authenticated session with the Client. If the End-User identified by the ID Token is logged in or is logged in by the request, then the Authorization Server returns a positive response. [string]	-
login_hint	Hint to the Authorization Server about the login identifier the End-User might use to log in (if necessary). [string]	-
acr_values	Requested Authentication Context Class Reference values. Space-separated string that specifies the acr values that the Authorization Server is being requested to use for processing this Authentication Request, with the values appearing in order of preference. [string]	-
amr_values	AMR Values. [string]	-
request	This parameter enables OpenID Connect requests to be passed in a single, self-contained parameter and to be optionally signed and/or encrypted. The parameter value is a Request Object value. It represents the request as a JWT whose Claims are the request parameters. [string]	-
request_uri	This parameter enables OpenID Connect requests to be passed by reference, rather than by value. The request_uri value is a URL using the https scheme referencing a resource containing a Request Object value, which is a JWT containing the request parameters. [string]	-
request_session_id	Request session id. [string]	-
session_id	Session id of this call. [string]	-
origin_headers	Origin headers. Used in custom workflows. [string]	-

Name	Description	Required
code_challenge	PKCE code challenge. [string]	-
code_challenge_method	PKCE code challenge method. [string]	-
custom_response_headers	Custom Response Headers. [string]	-
claims	Requested Claims. [string]	-

3.1.1.7.4. Return Type

-

3.1.1.7.5. Content Type

- application/json

3.1.1.7.6. Responses

Table 8. http response codes

Code	Message	Datatype
200	OK	<<>>
302	Error codes for authorization endpoint.	*AuthorizeError*
400	Invalid parameters are provided to endpoint.	*ErrorResponse*
401	Unauthorized access request.	*ErrorResponse*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.1.7.7. Samples

3.1.1.8. POST Clientinfo

`POST /clientinfo`

To get Claims details about the registered client.

3.1.1.8.1. Description

The ClientInfo Endpoint is an OAuth 2.0 Protected Resource that returns Claims about the registered client.

3.1.1.8.2. Parameters

3.1.1.8.3. Form Parameter

Name	Description	Required
access_token	Client-specific access token. [string]	X

Header Parameters

Name	Description	Required
Authorization		-

3.1.1.8.4. Return Type

ClientInfoResponse

3.1.1.8.5. Content Type

- application/json

3.1.1.8.6. Responses

Table 9. http response codes

Code	Message	Datatype
200	OK	*ClientInfoResponse*
400	Invalid Request are provided to endpoint.	[inline_response_400]

3.1.1.8.7. Samples

3.1.1.9. POST Introspection

`POST /introspection`

The Introspection OAuth 2 Endpoint.

3.1.1.9.1. Description

The Introspection OAuth 2 Endpoint.

3.1.1.9.2. Parameters

3.1.1.9.3. Form Parameter

Name	Description	Required
token	Client access token. [string]	X

Header Parameters

Name	Description	Required
Authorization	Client Authorization details that contains the access token along with other details.	X

3.1.1.9.4. Return Type

IntrospectionResponse

3.1.1.9.5. Content Type

- application/json

3.1.1.9.6. Responses

Table 10. http response codes

Code	Message	Datatype
200	OK	*IntrospectionResponse*
400	Error codes for introspection endpoint.	*AuthorizeError*
401	Unauthorized access request.	*ErrorResponse*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.1.9.7. Samples

3.1.1.10. POST Token

`POST /token`

To obtain an Access Token, an ID Token, and optionally a Refresh Token, the RP (Client).

3.1.1.10.1. Description

To obtain an Access Token, an ID Token, and optionally a Refresh Token, the RP (Client).

3.1.1.10.2. Parameters

3.1.1.10.3. Form Parameter

Name	Description	Required
grant_type	Provide a list of the OAuth 2.0 grant types that the Client is declaring that it will restrict itself to using. [String]	X
code	Code which is returned by authorization endpoint. (For grant_type\=authorization_code) [string]	-
redirect_uri	Redirection URI to which the response will be sent. This URI MUST exactly match one of the Redirection URI values for the Client pre-registered at the OpenID Provider. [string]	-
username	End-User username. [string]	-

Name	Description	Required
password	End-User password. [string]	-
scope	OpenID Connect requests MUST contain the openid scope value. If the openid scope value is not present, the behavior is entirely unspecified. Other scope values MAY be present. Scope values used that are not understood by an implementation SHOULD be ignored. [String]	-
assertion	Assertion. [string]	-
refresh_token	Refresh token. [string]	-
client_id	OAuth 2.0 Client Identifier valid at the Authorization Server. [string]	-
client_secret	The client secret. The client MAY omit the parameter if the client secret is an empty string. [string]	-
code_verifier	The client's PKCE code verifier. [string]	-
ticket	[string]	-
claim_token	[string]	-
claim_token_format	[string]	-
pct	[string]	-
rpt	[string]	-

3.1.1.10.4. Return Type

[inline_response_200]

3.1.1.10.5. Content Type

- application/json

3.1.1.10.6. Responses

Table 11. http response codes

Code	Message	Datatype
200	OK	[inline_response_200]
400	Invalid parameters provided to endpoint.	[inline_response_400_2]
401	Unauthorized access request.	*ErrorResponse*
403	Invalid details provided hence access denied.	*ErrorResponse*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.1.10.7. Samples

3.1.1.11. POST Userinfo

`POST /userinfo`

Returns Claims about the authenticated End-User.

3.1.1.11.1. Description

Returns Claims about the authenticated End-User.

3.1.1.11.2. Parameters

3.1.1.11.3. Form Parameter

Name	Description	Required
access_token	OAuth 2.0 Access Token. [string]	X

Header Parameters

Name	Description	Required
Authorization	Client Authorization details that contains the access token along with other details.	-

3.1.1.11.4. Return Type

[Object]

3.1.1.11.5. Content Type

- application/jwt
- application/json

3.1.1.11.6. Responses

Table 12. http response codes

Code	Message	Datatype
200	OK	[Object]
400	Invalid parameters provided to endpoint.	[inline_response_400_6]
401	Invalid parameters provided to endpoint.	[inline_response_401]
403	Invalid parameters provided to endpoint.	[inline_response_403]
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.1.11.7. Samples

3.1.1.12. revoke

`POST /revoke`

Revoke an Access Token or a Refresh Token, the RP (Client).

3.1.1.12.1. Description

Revoke an Access Token or a Refresh Token, the RP (Client).

3.1.1.12.2. Parameters

3.1.1.12.3. Form Parameter

Name	Description	Required
token	The token that the client wants to get revoked. [string]	X
token_type_hint	A hint about the type of the token submitted for revocation. [string]	-

3.1.1.12.4. Return Type

-

3.1.1.12.5. Content Type

- content
- application/json

3.1.1.12.6. Responses

Table 13. http response codes

Code	Message	Datatype
200	OK	<<>>
400	Invalid parameters provided to endpoint.	[inline_response_400_4]

3.1.1.12.7. Samples

3.1.1.13. revokeSession

`POST /revoke_session`

Revoke all sessions for user.

3.1.1.13.1. Description

Revoke all sessions for user (requires revoke_session scope).

3.1.1.13.2. Parameters

3.1.1.13.3. Form Parameter

Name	Description	Required
user_criterion_key	user criterion key (e.g. uid) [string]	X
user_criterion_value	user criterion value (e.g. chris) [string]	X

3.1.1.13.4. Return Type

-

3.1.1.13.5. Content Type

- application/json

3.1.1.13.6. Responses

Table 14. http response codes

Code	Message	Datatype
200	OK - Returned if request was processed successfully. Means it will return in case sessions are found as well as in case sessions are not found (error is not returned to not disclose internal information).	<<>>
401	Unauthorized access request.	*ErrorResponse*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.1.13.7. Samples

3.1.1.14. sessionStatus

`GET /session_status`

Determine current sesion status.

3.1.1.14.1. Description

Determine current sesion status.

3.1.1.14.2. Parameters

3.1.1.14.3. Return Type

SessionStateObject

3.1.1.14.4. Content Type

- application/json

3.1.1.14.5. Responses

Table 15. http response codes

Code	Message	Datatype
200	OK	*SessionStateObject*

3.1.1.14.6. Samples

3.1.2. UMA - Authorization

3.1.2.1. DELETE HostRsrcResourceSet

`DELETE /host/rsrc/resource_set/{rsid}`

Deletes a previously registered resource.

3.1.2.1.1. Description

Deletes a previously registered resource.

3.1.2.1.2. Parameters

Path Parameters

Name	Description	Required
rsid	Resource ID	X

Header Parameters

Name	Description	Required
Authorization		X

3.1.2.1.3. Return Type

-

3.1.2.1.4. Content Type

- application/json

3.1.2.1.5. Responses

Table 16. http response codes

Code	Message	Datatype
204	OK	<<>>
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.1.6. Samples

3.1.2.2. GET HostRsrcResourceSet

`GET /host/rsrc/resource_set`

Lists all previously registered resource.

3.1.2.2.1. Description

Lists all previously registered resource.

3.1.2.2.2. Parameters

Header Parameters

Name	Description	Required
Authorization		X

Query Parameters

Name	Description	Required
scope	Scope uri.	-

3.1.2.2.3. Return Type

[List]

3.1.2.2.4. Content Type

- application/json

3.1.2.2.5. Responses

Table 17. http response codes

Code	Message	Datatype
200	OK	List[[string]]
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.2.6. Samples

3.1.2.3. GET HostRsrcResourceSet/{rsid}

`GET /host/rsrc/resource_set/{rsid}`

Reads a previously registered resource.

3.1.2.3.1. Description

Reads a previously registered resource.

3.1.2.3.2. Parameters

Path Parameters

Name	Description	Required
rsid	Resource description ID.	X

Header Parameters

Name	Description	Required
Authorization	Client Authorization details that contains the access token along with other details.	X

3.1.2.3.3. Return Type

UmaResourceWithId

3.1.2.3.4. Content Type

- application/json

3.1.2.3.5. Responses

Table 18. http response codes

Code	Message	Datatype
200	OK	*UmaResourceWithId*
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.3.6. Samples

3.1.2.4. GET Introspection

`GET /introspection`

The Introspection OAuth 2 Endpoint.

3.1.2.4.1. Description

The Introspection OAuth 2 Endpoint.

3.1.2.4.2. Parameters

Header Parameters

Name	Description	Required
Authorization	Client Authorization details that contains the access token along with other details.	X

Query Parameters

Name	Description	Required
token		X
token_type_hint	ID Token previously issued by the Authorization Server being passed as a hint about the End-User.	-
response_as_jwt	OPTIONAL. Boolean value with default value false. If true, returns introspection response as JWT (signed based on client configuration used for authentication to Introspection Endpoint).	-

3.1.2.4.3. Return Type

IntrospectionResponse

3.1.2.4.4. Content Type

- application/json

3.1.2.4.5. Responses

Table 19. http response codes

Code	Message	Datatype
200	OK	*IntrospectionResponse*
400	Error codes for introspection endpoint.	*AuthorizeError*
401	Unauthorized access request.	*ErrorResponse*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.2.4.6. Samples

3.1.2.5. GET RptStatus

`GET /rpt/status`

The Introspection OAuth 2 Endpoint for RPT.

3.1.2.5.1. Description

The Introspection OAuth 2 Endpoint for RPT.

3.1.2.5.2. Parameters

Header Parameters

Name	Description	Required
Authorization		X

Query Parameters

Name	Description	Required
token		X
token_type_hint		-

3.1.2.5.3. Return Type

RptIntrospectionResponse

3.1.2.5.4. Content Type

- application/json

3.1.2.5.5. Responses

Table 20. http response codes

Code	Message	Datatype
200	OK	*RptIntrospectionResponse*
405	Introspection of RPT is not allowed.	*ErrorResponse*
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.5.6. Samples

3.1.2.6. GET UmaGatherClaims

`GET /uma/gather_claims`

UMA Claims Gathering Endpoint.

3.1.2.6.1. Description

UMA Claims Gathering Endpoint.

3.1.2.6.2. Parameters

Query Parameters

Name	Description	Required
client_id	OAuth 2.0 Client Identifier valid at the Authorization Server.	-
ticket		-
claims_redirect_uri		-
state		-
reset		-
authentication		-

3.1.2.6.3. Return Type

-

3.1.2.6.4. Content Type

- application/json

3.1.2.6.5. Responses

Table 21. http response codes

Code	Message	Datatype
302	Resource Found.	<<>>
400	Invalid parameters provided to endpoint.	[inline_response_400_5]
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.6.6. Samples

3.1.2.7. hostRsrcPr

`POST /host/rsrc_pr`

Registers permission.

3.1.2.7.1. Description

Registers permission.

3.1.2.7.2. Parameters

3.1.2.7.3. Form Parameter

Name	Description	Required
resource_id	The identifier for a resource to which this client is seeking access. The identifier MUST correspond to a resource that was previously registered. [string]	X
resource_scopes	An array referencing zero or more strings representing scopes to which access was granted for this resource. Each string MUST correspond to a scope that was registered by this resource server for the referenced resource. [String]	X

Header Parameters

Name	Description	Required
Authorization	Client Authorization details that contains the access token along with other details.	X

3.1.2.7.4. Return Type

array[[inline_response_201]]

3.1.2.7.5. Content Type

- application/json

3.1.2.7.6. Responses

Table 22. http response codes

Code	Message	Datatype
201	OK	List[[inline_response_201]]
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.7.7. Samples

3.1.2.8. jwks

`GET /jwks`

A JSON Web Key (JWK) used by server. JWK is a JSON data structure that represents a set of public keys as a JSON object [RFC4627].

3.1.2.8.1. Description

Provides list of JWK used by server.

3.1.2.8.2. Parameters

3.1.2.8.3. Return Type

WebKeysConfiguration

3.1.2.8.4. Content Type

- application/json

3.1.2.8.5. Responses

Table 23. http response codes

Code	Message	Datatype
200	OK	*WebKeysConfiguration*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.2.8.6. Samples

3.1.2.9. POST HostRsrcResourceSet

`POST /host/rsrc/resource_set`

Adds a new resource description.

3.1.2.9.1. Description

Adds a new resource description.

3.1.2.9.2. Parameters

3.1.2.9.3. Body Parameter

Name	Description	Required
UmaResource	*UmaResource* UmaResource	-

Header Parameters

Name	Description	Required
Authorization	Client Authorization details that contains the access token along with other details.	X

3.1.2.9.4. Return Type

UmaResourceResponse

3.1.2.9.5. Content Type

- application/json

3.1.2.9.6. Responses

Table 24. http response codes

Code	Message	Datatype
201	OK	*UmaResourceResponse*
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.9.7. Samples

3.1.2.10. POST Introspection

`POST /introspection`

The Introspection OAuth 2 Endpoint.

3.1.2.10.1. Description

The Introspection OAuth 2 Endpoint.

3.1.2.10.2. Parameters

3.1.2.10.3. Form Parameter

Name	Description	Required
token	Client access token. [string]	X

Header Parameters

Name	Description	Required
Authorization	Client Authorization details that contains the access token along with other details.	X

3.1.2.10.4. Return Type

IntrospectionResponse

3.1.2.10.5. Content Type

- application/json

3.1.2.10.6. Responses

Table 25. http response codes

Code	Message	Datatype
200	OK	*IntrospectionResponse*
400	Error codes for introspection endpoint.	*AuthorizeError*
401	Unauthorized access request.	*ErrorResponse*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.2.10.7. Samples

3.1.2.11. POST RptStatus

`POST /rpt/status`

The Introspection OAuth 2 Endpoint for RPT.

3.1.2.11.1. Description

The Introspection OAuth 2 Endpoint for RPT.

3.1.2.11.2. Parameters

3.1.2.11.3. Form Parameter

Name	Description	Required
token	Client access token. [string]	X
token_type_hint	ID Token previously issued by the Authorization Server being passed as a hint about the End-User. [string]	-

Header Parameters

Name	Description	Required
Authorization	Client Authorization details that contains the access token along with other details.	X

3.1.2.11.4. Return Type

[RptIntrospectionResponse_1]

3.1.2.11.5. Content Type

- application/json

3.1.2.11.6. Responses

Table 26. http response codes

Code	Message	Datatype
200	OK	[RptIntrospectionResponse_1]
405	Introspection of RPT is not allowed.	*ErrorResponse*
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.11.7. Samples

3.1.2.12. POST Token

`POST /token`

To obtain an Access Token, an ID Token, and optionally a Refresh Token, the RP (Client).

3.1.2.12.1. Description

To obtain an Access Token, an ID Token, and optionally a Refresh Token, the RP (Client).

3.1.2.12.2. Parameters

3.1.2.12.3. Form Parameter

Name	Description	Required
grant_type	Provide a list of the OAuth 2.0 grant types that the Client is declaring that it will restrict itself to using. [String]	X
code	Code which is returned by authorization endpoint. (For grant_type\=authorization_code) [string]	-
redirect_uri	Redirection URI to which the response will be sent. This URI MUST exactly match one of the Redirection URI values for the Client pre-registered at the OpenID Provider. [string]	-
username	End-User username. [string]	-
password	End-User password. [string]	-
scope	OpenID Connect requests MUST contain the openid scope value. If the openid scope value is not present, the behavior is entirely unspecified. Other scope values MAY be present. Scope values used that are not understood by an implementation SHOULD be ignored. [String]	-
assertion	Assertion. [string]	-

Name	Description	Required
refresh_token	Refresh token. [string]	-
client_id	OAuth 2.0 Client Identifier valid at the Authorization Server. [string]	-
client_secret	The client secret. The client MAY omit the parameter if the client secret is an empty string. [string]	-
code_verifier	The client's PKCE code verifier. [string]	-
ticket	[string]	-
claim_token	[string]	-
claim_token_format	[string]	-
pct	[string]	-
rpt	[string]	-

3.1.2.12.4. Return Type

[inline_response_200]

3.1.2.12.5. Content Type

- application/json

3.1.2.12.6. Responses

Table 27. http response codes

Code	Message	Datatype
200	OK	[inline_response_200]
400	Invalid parameters provided to endpoint.	[inline_response_400_2]
401	Unauthorized access request.	*ErrorResponse*
403	Invalid details provided hence access denied.	*ErrorResponse*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.2.12.7. Samples

3.1.2.13. POST UmaGatherClaims

`POST /uma/gather_claims`

UMA Claims Gathering Endpoint

3.1.2.13.1. Description

UMA Claims Gathering Endpoint

3.1.2.13.2. Parameters

3.1.2.13.3. Form Parameter

Name	Description	Required
client_id	OAuth 2.0 Client Identifier valid at the Authorization Server. [string]	-
ticket	[string]	-
claims_redirect_uri	[string]	-
state	[string]	-
reset	[boolean]	-
authentication	[boolean]	-

3.1.2.13.4. Return Type

-

3.1.2.13.5. Content Type

- application/json

3.1.2.13.6. Responses

Table 28. http response codes

Code	Message	Datatype
302	Resource Found.	<<>>
400	Invalid parameters provided to endpoint.	[inline_response_400_5]
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.13.7. Samples

3.1.2.14. PUT HostRsrcResourceSet{rsid}

`PUT /host/rsrc/resource_set/{rsid}`

Updates a previously registered resource.

3.1.2.14.1. Description

Updates a previously registered resource.

3.1.2.14.2. Parameters

Path Parameters

Name	Description	Required
rsid	Resource ID.	X

3.1.2.14.3. Body Parameter

Name	Description	Required
UmaResource1	*UmaResource1* UmaResource	-

Header Parameters

Name	Description	Required
Authorization		X

3.1.2.14.4. Return Type

UmaResourceResponse

3.1.2.14.5. Content Type

- application/json

3.1.2.14.6. Responses

Table 29. http response codes

Code	Message	Datatype
200	OK	*UmaResourceResponse*
404	Invalid parameters provided to endpoint.	[inline_response_404]
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.14.7. Samples

3.1.2.15. revoke

`POST /revoke`

Revoke an Access Token or a Refresh Token, the RP (Client).

3.1.2.15.1. Description

Revoke an Access Token or a Refresh Token, the RP (Client).

3.1.2.15.2. Parameters

3.1.2.15.3. Form Parameter

Name	Description	Required
token	The token that the client wants to get revoked. [string]	X
token_type_hint	A hint about the type of the token submitted for revocation. [string]	-

3.1.2.15.4. Return Type

-

3.1.2.15.5. Content Type

- content
- application/json

3.1.2.15.6. Responses

Table 30. http response codes

Code	Message	Datatype
200	OK	<<>>
400	Invalid parameters provided to endpoint.	[inline_response_400_4]

3.1.2.15.7. Samples

3.1.2.16. uma2Configuration

`GET /uma2-configuration`

Gets UMA configuration data.

3.1.2.16.1. Description

Gets UMA configuration data.

3.1.2.16.2. Parameters

3.1.2.16.3. Return Type

[inline_response_200_1]

3.1.2.16.4. Content Type

- application/json

3.1.2.16.5. Responses

Table 31. http response codes

Code	Message	Datatype
200	OK	[inline_response_200_1]
500	Invalid parameters provided to endpoint.	[inline_response_500]

3.1.2.16.6. Samples

3.1.3. Registration

3.1.3.1. DELETE Register

`DELETE /register`

Deletes the client info for a previously registered client.

3.1.3.1.1. Description

The Client Registration Endpoint removes the Client Metadata for a previously registered client.

3.1.3.1.2. Parameters

Header Parameters

Name	Description	Required
Authorization	Authorization header carrying \\"registration_access_token\\" issued before as a Bearer token	X

Query Parameters

Name	Description	Required
client_id	Client ID that identifies client.	X

3.1.3.1.3. Return Type

-

3.1.3.1.4. Content Type

- application/json

3.1.3.1.5. Responses

Table 32. http response codes

Code	Message	Datatype
204	OK	<<>>
400	Invalid parameters provided to endpoint.	[inline_response_400_1]
401	Invalid parameters provided to endpoint.	[inline_response_400_1]
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.3.1.6. Samples

3.1.3.2. GET Register

`GET /register`

Get client information for a previously registered client.

3.1.3.2.1. Description

Get client information for a previously registered client.

3.1.3.2.2. Parameters

Header Parameters

Name	Description	Required
Authorization	Authorization header carrying \\"registration_access_token\\" issued before as a Bearer token	X

Query Parameters

Name	Description	Required
client_id	Client ID that identifies client.	X

3.1.3.2.3. Return Type

ClientResponse

3.1.3.2.4. Content Type

- application/json

3.1.3.2.5. Responses

Table 33. http response codes

Code	Message	Datatype
200	OK	*ClientResponse*
400	Invalid parameters provided to endpoint.	[inline_response_400_1]
401	Invalid parameters are provided to endpoint.	*ErrorResponse*
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.3.2.6. Samples

3.1.3.3. POST Register

`POST /register`

Registers new client dynamically.

3.1.3.3.1. Description

The Client Registration Endpoint is an OAuth 2.0 Protected Resource through which a new Client registration can be requested.

3.1.3.3.2. Parameters

3.1.3.3.3. Body Parameter

Name	Description	Required
RegisterParams1	*RegisterParams1* RegisterParams	-

3.1.3.3.4. Return Type

RegisterResponseParam

3.1.3.3.5. Content Type

- application/json

3.1.3.3.6. Responses

Table 34. http response codes

Code	Message	Datatype
200	OK	*RegisterResponseParam*
400	Invalid parameters provided to endpoint.	[inline_response_400_3]
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.3.3.7. Samples

3.1.3.4. PUT Register

`PUT /register`

Updates Client Metadata for a registered client.

3.1.3.4.1. Description

Updates Client Metadata for a registered client.

3.1.3.4.2. Parameters

3.1.3.4.3. Body Parameter

Name	Description	Required
RegisterParams	*RegisterParams* RegisterParams	-

Header Parameters

Name	Description	Required
Authorization	Authorization header carrying \\"registration_access_token\\" issued before as a Bearer token	X

Query Parameters

Name	Description	Required
client_id	Client ID that identifies client that must be updated by this request.	X

3.1.3.4.4. Return Type

RegisterResponseParam

3.1.3.4.5. Content Type

- application/json

3.1.3.4.6. Responses

Table 35. http response codes

Code	Message	Datatype
200	OK	*RegisterResponseParam*
400	Invalid parameters provided to endpoint.	[inline_response_400_2]

Code	Message	Datatype
500	Internal error occured. Please check log file for details.	*ErrorResponse*

3.1.3.4.7. Samples

3.2. Models

3.2.1. *AuthorizeError*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.2. *ClientInfoResponse*

Client details in response.

Field Name	Required	Type	Description	Format
displayName		String		
inum		String	XRI i-number	
oxAuthAppType		String	oxAuth Appication type	
oxAuthIdTokenSignedResponseAlg		String	oxAuth ID Token Signed Response Algorithm	
oxAuthRedirectURI		List of [string]	Array of redirect URIs values used in the Authorization	
oxId		String	oxAuth Attribute Scope Id	
custom_attributes		List of [string]		

3.2.3. *ClientResponse*

Field Name	Required	Type	Description	Format
redirect_uris		List of [string]	Redirection URI values used by the Client. One of these registered Redirection URI values must exactly match the redirect_uri parameter value used in each Authorization Request	

Field Name	Required	Type	Description	Format
claims_redirect_uri		List of [string]	Array of The Claims Redirect URIs to which the client wishes the authorization server to direct the requesting party's user agent after completing its interaction.	
response_types		List of [string]	A list of the OAuth 2.0 response_type values that the Client is declaring that it will restrict itself to using. If omitted, the default is that the Client will use only the code Response Type. Allowed values are code, token, id_token.	
grant_types		List of [string]	A list of the OAuth 2.0 Grant Types that the Client is declaring that it will restrict itself to using.	
contacts		List of [string]	e-mail addresses of people responsible for this Client.	
client_name		String	Name of the Client to be presented to the user.	
logo_uri		String	URL that references a logo for the Client application	
client_uri		String	URL of the home page of the Client. The value of this field must point to a valid Web page.	
policy_uri		String	URL that the Relying Party Client provides to the End-User to read about the how the profile data will be used.	
tos_uri		String	URL that the Relying Party Client provides to the End-User to read about the Relying Party's terms of service.	
jwks_uri		String	URL for the Client's JSON Web Key Set (JWK) document containing key(s) that are used for signing requests to the OP. The JWK Set may also contain the Client's encryption keys(s) that are used by the OP to encrypt the responses to the Client. When both signing and encryption keys are made available, a use (Key Use) parameter value is required for all keys in the document to indicate each key's intended usage .	

Field Name	Required	Type	Description	Format
jwks		String	Client's JSON Web Key Set (JWK) document, passed by value. The semantics of the jwks parameter are the same as the jwks_uri parameter, other than that the JWK Set is passed by value, rather than by reference. This parameter is intended only to be used by Clients that, for some reason, are unable to use the jwks_uri parameter, for instance, by native applications that might not have a location to host the contents of the JWK Set. If a Client can use jwks_uri, it must not use jwks. One significant downside of jwks is that it does not enable key rotation. The jwks_uri and jwks parameters must not be used together.	
sector_identifier_uri		String	URL using the https scheme to be used in calculating Pseudonymous Identifiers by the OP.	
subject_type		String	Subject type requested for the Client ID. Valid types include pairwise and public.	
rpt_as_jwt		Boolean	Specifies whether RPT should be return as signed JWT.	
access_token_as_jwt		Boolean	Specifies whether access token as signed JWT.	
access_token_signing_alg		String	Specifies signing algorithm that has to be used during JWT signing. If it's not specified, then the default OP signing algorithm will be used .	
id_token_signed_response_alg		String	JWS alg algorithm (JWA) required for signing the ID Token issued to this Client.	
id_token_encrypted_response_alg		String	JWE alg algorithm (JWA) required for encrypting the ID Token issued to this Client.	
id_token_encrypted_response_enc		String	JWE enc algorithm (JWA) required for encrypting the ID Token issued to this Client.	

Field Name	Required	Type	Description	Format
userinfo_signed_response_alg		String	JWS alg algorithm (JWA) required for signing UserInfo Responses.	
userinfo_encrypted_response_alg		String	JWE alg algorithm (JWA) required for encrypting UserInfo Responses.	
userinfo_encrypted_response_enc		String	JWE enc algorithm (JWA) required for encrypting UserInfo Responses.	
request_object_signing_alg		String	JWS alg algorithm (JWA) that must be used for signing Request Objects sent to the OP.	
request_object_encryption_alg		String	JWE alg algorithm (JWA) the RP is declaring that it may use for encrypting Request Objects sent to the OP.	
request_object_encryption_enc		String	JWE enc algorithm (JWA) the RP is declaring that it may use for encrypting Request Objects sent to the OP.	
token_endpoint_auth_method		String	Requested Client Authentication method for the Token Endpoint.	
token_endpoint_auth_signing_alg		String	JWS alg algorithm (JWA) that must be used for signing the JWT used to authenticate the Client at the Token Endpoint for the private_key_jwt and client_secret_jwt authentication methods.	
default_max_age		Integer	Specifies the Default Maximum Authentication Age.	
require_auth_time		Boolean	Boolean value specifying whether the auth_time Claim in the ID Token is required. It is required when the value is true.	
default_acr_values		List of [string]	Array of default requested Authentication Context Class Reference values that the Authorization Server must use for processing requests from the Client.	
initiate_login_uri		String	Specifies the URI using the https scheme that the authorization server can call to initiate a login at the client.	

Field Name	Required	Type	Description	Format
post_logout_redirect_uris		List of [string]	Provide the URLs supplied by the RP to request that the user be redirected to this location after a logout has been performed.	
frontchannel_logout_uri		String	RP URL that will cause the RP to log itself out when rendered in an iframe by the OP.	
frontchannel_logout_session_required		Boolean	Boolean value specifying whether the RP requires that a session ID query parameter be included to identify the RP session at the OP when the logout_uri is used. If omitted, the default value is false.	
backchannel_logout_uri		String	RP URL that will cause the RP to log itself out when sent a Logout Token by the OP.	
backchannel_logout_session_required		Boolean	Boolean value specifying whether the RP requires that a session ID Claim be included in the Logout Token to identify the RP session with the OP when the backchannel_logout_uri is used. If omitted, the default value is false.	
request_uris		List of [string]	Provide a list of request_uri values that are pre-registered by the Client for use at the Authorization Server.	
scopes		String	This param will be removed in a future version because the correct is 'scope' not 'scopes', see (rfc7591).	
claims		String	String containing a space-separated list of claims that can be requested individually.	
id_token_token_binding_cnf		String	Specifies the JWT Confirmation Method member name (e.g. tbh) that the Relying Party expects when receiving Token Bound ID Tokens. The presence of this parameter indicates that the Relying Party supports Token Binding of ID Tokens. If omitted, the default is that the Relying Party does not support Token Binding of ID Tokens.	

Field Name	Required	Type	Description	Format
tls_client_auth_subject_dn		String	An string representation of the expected subject distinguished name of the certificate, which the OAuth client will use in mutual TLS authentication.	
allow_spontaneous_scopes		Boolean	Specifies whether to allow spontaneous scopes for client. The default value is false.	
spontaneous_scopes		List of [string]	List of spontaneous scopes	
run_introspection_script_before_access_token_as_jwt_creation_and_include_claims		Boolean	Boolean value with default value false. If true and access_token_as_jwt\=true then run introspection script and transfer claims into JWT.	
keep_client_authorization_after_expiration		Boolean	Boolean value indicating if the client authorization will not be removed afer expiration (expiration date is same as client's expiration that created it). The default value is false.	
scope		List of [string]	Provide list of scope which are used during authentication to authorize access to resource.	
authorized_origins		List of [string]	specifies authorized JavaScript origins.	
access_token_lifetime		Integer	Specifies the Client-specific access token expiration.	
software_id		String	Specifies a unique identifier string (UUID) assigned by the client developer or software publisher used by registration endpoints to identify the client software to be dynamically registered.	
software_version		String	Specifies a version identifier string for the client software identified by 'software_id'. The value of the 'software_version' should change on any update to the client software identified by the same 'software_id'.	

Field Name	Required	Type	Description	Format
software_statement		String	specifies a software statement containing client metadata values about the client software as claims. This is a string value containing the entire signed JWT.	
backchannel_token_delivery_mode		String	specifies how backchannel token will be deliveried.	
backchannel_client_notification_endpoint		String	Client Initiated Backchannel Authentication (CIBA) enables a Client to initiate the authentication of an end-user by means of out-of-band mechanisms. Upon receipt of the notification, the Client makes a request to the token endpoint to obtain the tokens.	
backchannel_authentication_request_signing_alg		String	The JWS algorithm alg value that the Client will use for signing authentication request, as described in Section 7.1.1. of OAuth 2.0 [RFC6749]. When omitted, the Client will not send signed authentication requests.	
backchannel_user_code_parameter		Boolean	Boolean value specifying whether the Client supports the user_code parameter. If omitted, the default value is false.	

3.2.4. *EndSessionError*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.5. *ErrorResponse*

Field Name	Required	Type	Description	Format
error	X	String		
error_description	X	String		

Field Name	Required	Type	Description	Format
details		String		

3.2.6. *InlineResponse200*

AccessTokenResponse.

Field Name	Required	Type	Description	Format
access_token	X	String	The access token issued by the authorization server.	
token_type	X	String	The access token type provides the client with the information required to successfully utilize the access token to make a protected resource request (along with type-specific attributes).	
expires_in		Integer	The lifetime in seconds of the access token. For example, the value \\"3600\\" denotes that the access token will expire in one hour from the time the response was generated.	
refresh_token		String	The refresh token, which can be used to obtain new access tokens using the same authorization grant	
scope		List of [string]		
id_token		String		

3.2.7. *InlineResponse2001*

UmaMetadata

Field Name	Required	Type	Description	Format
issuer	X	String	The authorization server’s issuer identifier	
authorization_endpoint	X	String	URL of the authorization server	

Field Name	Required	Type	Description	Format
uma_profiles_supported		List of [string]	UMA profiles supported by this authorization server. The value is an array of string values, where each string value is a URI identifying an UMA profile	
permission_endpoint		String	The endpoint URI at which the resource server requests permissions on the client's behalf.	
resource_registration_endpoint		String	The endpoint URI at which the resource server registers resources to put them under authorization manager protection.	
scope_endpoint		String	The Scope endpoint URI.	

3.2.8. *InlineResponse201*

Field Name	Required	Type	Description	Format
resource_id	X	String	The identifier for a resource to which this client is seeking access. The identifier MUST correspond to a resource that was previously registered.	
resource_scopes	X	List of [string]	An array referencing zero or more strings representing scopes to which access was granted for this resource. Each string MUST correspond to a scope that was registered by this resource server for the referenced resource.	
params		Map of [string]	A key/value map that can contain custom parameters.	
exp		Long	Number of seconds since January 1 1970 UTC, indicating when this token will expire.	int64

3.2.9. *InlineResponse400*

Field Name	Required	Type	Description	Format
error	X	String		enum

Field Name	Required	Type	Description	Format
error_description	X	String		
details		String		

3.2.10. *InlineResponse4001*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.11. *InlineResponse4002*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.12. *InlineResponse4003*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.13. *InlineResponse4004*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.14. *InlineResponse4005*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.15. *InlineResponse4006*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.16. *InlineResponse401*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.17. *InlineResponse403*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.18. *InlineResponse404*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.19. *InlineResponse500*

Field Name	Required	Type	Description	Format
error	X	String		enum
error_description	X	String		
details		String		

3.2.20. *IntrospectionResponse*

meta-information about token

Field Name	Required	Type	Description	Format
active	X	Boolean	Boolean indicator of whether or not the presented token is currently active.	
scope		List of [string]	Provide list of scopes to which access was granted for this resource.	
client_id		String	Client identifier for the OAuth 2.0 client that requested this token.	
username		String	Human-readable identifier for the resource owner who authorized this token.	
token_type		String	Type of the token as defined in Section 5.1 of OAuth 2.0 [RFC6749].	
exp		Integer	Integer timestamp, measured in the number of seconds since January 1 1970 UTC, indicating when this permission will expire.	
iat		Integer		
sub		String	Subject of the token, as defined in JWT [RFC7519].	
aud		String	Service-specific string identifier or list of string identifiers representing the intended audience for this token, as defined in JWT [RFC7519].	
iss		String	String representing the issuer of this token, as defined in JWT [RFC7519].	
acr_values		String	Authentication Context Class Reference values.	

Field Name	Required	Type	Description	Format
jti		String	String identifier for the token, as defined in JWT.	

3.2.21. *JsonWebKey*

Field Name	Required	Type	Description	Format
kid	X	String		
kty	X	String		
use	X	String		
alg	X	String		
crv		String		
exp	X	Long		int64
x5c	X	List of [string]		
n		String		
e		String		
x		String		
y		String		

3.2.22. *RegisterParams* **RegisterParams**

Field Name	Required	Type	Description	Format
redirect_uris	X	List of [string]	Redirection URI values used by the Client. One of these registered Redirection URI values must exactly match the redirect_uri parameter value used in each Authorization Request	
claims_redirect_uri		List of [string]	Array of The Claims Redirect URIs to which the client wishes the authorization server to direct the requesting party's user agent after completing its interaction.	

Field Name	Required	Type	Description	Format
response_types		List of [string]	A list of the OAuth 2.0 response_type values that the Client is declaring that it will restrict itself to using. If omitted, the default is that the Client will use only the code Response Type. Allowed values are code, token, id_token.	
grant_types		List of [string]	A list of the OAuth 2.0 Grant Types that the Client is declaring that it will restrict itself to using.	
contacts		List of [string]	e-mail addresses of people responsible for this Client.	
client_name		String	Name of the Client to be presented to the user.	
logo_uri		String	URL that references a logo for the Client application	
client_uri		String	URL of the home page of the Client. The value of this field must point to a valid Web page.	
policy_uri		String	URL that the Relying Party Client provides to the End-User to read about the how the profile data will be used.	
tos_uri		String	URL that the Relying Party Client provides to the End-User to read about the Relying Party's terms of service.	
jwks_uri		String	URL for the Client's JSON Web Key Set (JWK) document containing key(s) that are used for signing requests to the OP. The JWK Set may also contain the Client's encryption keys(s) that are used by the OP to encrypt the responses to the Client. When both signing and encryption keys are made available, a use (Key Use) parameter value is required for all keys in the document to indicate each key's intended usage .	
jwks		List of *JsonWebKey*	List of JSON Web Key (JWK) - A JSON object that represents a cryptographic key. The members of the object represent properties of the key, including its value.	

Field Name	Required	Type	Description	Format
sector_identifier_uri		String	URL using the https scheme to be used in calculating Pseudonymous Identifiers by the OP.	
subject_type		String	Subject type requested for the Client ID. Valid types include pairwise and public.	
rpt_as_jwt		Boolean	Specifies whether RPT should be return as signed JWT.	
access_token_as_jwt		Boolean	Specifies whether access token as signed JWT.	
access_token_signing_alg		String	Specifies signing algorithm that has to be used during JWT signing. If it's not specified, then the default OP signing algorithm will be used .	
id_token_signed_response_alg		String	JWS alg algorithm (JWA) required for signing the ID Token issued to this Client.	
id_token_encrypted_response_alg		String	JWE alg algorithm (JWA) required for encrypting the ID Token issued to this Client.	
id_token_encrypted_response_enc		String	JWE enc algorithm (JWA) required for encrypting the ID Token issued to this Client.	
userinfo_signed_response_alg		String	JWS alg algorithm (JWA) required for signing UserInfo Responses.	
userinfo_encrypted_response_alg		String	JWE alg algorithm (JWA) required for encrypting UserInfo Responses.	
userinfo_encrypted_response_enc		String	JWE enc algorithm (JWA) required for encrypting UserInfo Responses.	
request_object_signing_alg		String	JWS alg algorithm (JWA) that must be used for signing Request Objects sent to the OP.	
request_object_encryption_alg		String	JWE alg algorithm (JWA) the RP is declaring that it may use for encrypting Request Objects sent to the OP.	
request_object_encryption_enc		String	JWE enc algorithm (JWA) the RP is declaring that it may use for encrypting Request Objects sent to the OP.	

Field Name	Required	Type	Description	Format
token_endpoint_auth_method		String	Requested Client Authentication method for the Token Endpoint.	
token_endpoint_auth_signing_alg		String	JWS alg algorithm (JWA) that must be used for signing the JWT used to authenticate the Client at the Token Endpoint for the private_key_jwt and client_secret_jwt authentication methods.	
default_max_age		Integer	Specifies the Default Maximum Authentication Age.	
require_auth_time		Boolean	Boolean value specifying whether the auth_time Claim in the ID Token is required. It is required when the value is true.	
default_acr_values		List of [string]	Array of default requested Authentication Context Class Reference values that the Authorization Server must use for processing requests from the Client.	
initiate_login_uri		String	Specifies the URI using the https scheme that the authorization server can call to initiate a login at the client.	
post_logout_redirect_uris		List of [string]	Provide the URLs supplied by the RP to request that the user be redirected to this location after a logout has been performed.	
frontchannel_logout_uri		String	RP URL that will cause the RP to log itself out when rendered in an iframe by the OP.	
frontchannel_logout_session_required		Boolean	Boolean value specifying whether the RP requires that a session ID query parameter be included to identify the RP session at the OP when the logout_uri is used. If omitted, the default value is false.	
backchannel_logout_uri		String	RP URL that will cause the RP to log itself out when sent a Logout Token by the OP.	

Field Name	Required	Type	Description	Format
backchannel_logout_session_required		Boolean	Boolean value specifying whether the RP requires that a session ID Claim be included in the Logout Token to identify the RP session with the OP when the backchannel_logout_uri is used. If omitted, the default value is false.	
request_uris		List of [string]	Provide a list of request_uri values that are pre-registered by the Client for use at the Authorization Server.	
scopes		String	This param will be removed in a future version because the correct is 'scope' not 'scopes', see (rfc7591).	
claims		String	String containing a space-separated list of claims that can be requested individually.	
id_token_token_binding_cnf		String	Specifies the JWT Confirmation Method member name (e.g. tbh) that the Relying Party expects when receiving Token Bound ID Tokens. The presence of this parameter indicates that the Relying Party supports Token Binding of ID Tokens. If omitted, the default is that the Relying Party does not support Token Binding of ID Tokens.	
tls_client_auth_subject_dn		String	An string representation of the expected subject distinguished name of the certificate, which the OAuth client will use in mutual TLS authentication.	
allow_spontaneous_scopes		Boolean	Specifies whether to allow spontaneous scopes for client. The default value is false.	
spontaneous_scopes		List of [string]	List of spontaneous scopes	
run_introspection_script_before_access_token_as_jwt_creation_and_include_claims		Boolean	Boolean value with default value false. If true and access_token_as_jwt\=true then run introspection script and transfer claims into JWT.	

Field Name	Required	Type	Description	Format
keep_client_authorization_after_expiration		Boolean	Boolean value indicating if the client authorization will not be removed afer expiration (expiration date is same as client's expiration that created it). The default value is false.	
scope		List of [string]	Provide list of scope which are used during authentication to authorize access to resource.	
authorized_origins		List of [string]	specifies authorized JavaScript origins.	
access_token_lifetime		Integer	Specifies the Client-specific access token expiration.	
software_id		String	Specifies a unique identifier string (UUID) assigned by the client developer or software publisher used by registration endpoints to identify the client software to be dynamically registered.	
software_version		String	Specifies a version identifier string for the client software identified by 'software_id'. The value of the 'software_version' should change on any update to the client software identified by the same 'software_id'.	
software_statement		String	specifies a software statement containing client metadata values about the client software as claims. This is a string value containing the entire signed JWT.	
backchannel_token_delivery_mode		String	specifies how backchannel token will be deliveried.	
backchannel_client_notification_endpoint		String	Client Initiated Backchannel Authentication (CIBA) enables a Client to initiate the authentication of an end-user by means of out-of-band mechanisms. Upon receipt of the notification, the Client makes a request to the token endpoint to obtain the tokens.	

Field Name	Required	Type	Description	Format
backchannel_authentication_request_signing_alg		String	The JWS algorithm alg value that the Client will use for signing authentication request, as described in Section 7.1.1. of OAuth 2.0 [RFC6749]. When omitted, the Client will not send signed authentication requests.	
backchannel_user_code_parameter		Boolean	Boolean value specifying whether the Client supports the user_code parameter. If omitted, the default value is false.	
additional_audience		List of [string]	Additional audiences.	

3.2.23. *RegisterParams1* RegisterParams

Field Name	Required	Type	Description	Format
redirect_uris	X	List of [string]	Redirection URI values used by the Client. One of these registered Redirection URI values must exactly match the redirect_uri parameter value used in each Authorization Request	
claims_redirect_uri		List of [string]	Array of The Claims Redirect URIs to which the client wishes the authorization server to direct the requesting party's user agent after completing its interaction.	
response_types		List of [string]	A list of the OAuth 2.0 response_type values that the Client is declaring that it will restrict itself to using. If omitted, the default is that the Client will use only the code Response Type. Allowed values are code, token, id_token.	
grant_types		List of [string]	A list of the OAuth 2.0 Grant Types that the Client is declaring that it will restrict itself to using.	
contacts		List of [string]	e-mail addresses of people responsible for this Client.	
client_name		String	Name of the Client to be presented to the user.	

Field Name	Required	Type	Description	Format
logo_uri		String	URL that references a logo for the Client application	
client_uri		String	URL of the home page of the Client. The value of this field must point to a valid Web page.	
policy_uri		String	URL that the Relying Party Client provides to the End-User to read about the how the profile data will be used.	
tos_uri		String	URL that the Relying Party Client provides to the End-User to read about the Relying Party's terms of service.	
jwks_uri		String	URL for the Client's JSON Web Key Set (JWK) document containing key(s) that are used for signing requests to the OP. The JWK Set may also contain the Client's encryption keys(s) that are used by the OP to encrypt the responses to the Client. When both signing and encryption keys are made available, a use (Key Use) parameter value is required for all keys in the document to indicate each key's intended usage .	
jwks		List of *JsonWebKey*	List of JSON Web Key (JWK) - A JSON object that represents a cryptographic key. The members of the object represent properties of the key, including its value.	
sector_identifier_uri		String	URL using the https scheme to be used in calculating Pseudonymous Identifiers by the OP.	
subject_type		String	Subject type requested for the Client ID. Valid types include pairwise and public.	
rpt_as_jwt		Boolean	Specifies whether RPT should be return as signed JWT.	
access_token_as_jwt		Boolean	Specifies whether access token as signed JWT.	

Field Name	Required	Type	Description	Format
access_token_signing_alg		String	Specifies signing algorithm that has to be used during JWT signing. If it's not specified, then the default OP signing algorithm will be used .	
id_token_signed_response_alg		String	JWS alg algorithm (JWA) required for signing the ID Token issued to this Client.	
id_token_encrypted_response_alg		String	JWE alg algorithm (JWA) required for encrypting the ID Token issued to this Client.	
id_token_encrypted_response_enc		String	JWE enc algorithm (JWA) required for encrypting the ID Token issued to this Client.	
userinfo_signed_response_alg		String	JWS alg algorithm (JWA) required for signing UserInfo Responses.	
userinfo_encrypted_response_alg		String	JWE alg algorithm (JWA) required for encrypting UserInfo Responses.	
userinfo_encrypted_response_enc		String	JWE enc algorithm (JWA) required for encrypting UserInfo Responses.	
request_object_signing_alg		String	JWS alg algorithm (JWA) that must be used for signing Request Objects sent to the OP.	
request_object_encryption_alg		String	JWE alg algorithm (JWA) the RP is declaring that it may use for encrypting Request Objects sent to the OP.	
request_object_encryption_enc		String	JWE enc algorithm (JWA) the RP is declaring that it may use for encrypting Request Objects sent to the OP.	
token_endpoint_auth_method		String	Requested Client Authentication method for the Token Endpoint.	
token_endpoint_auth_signing_alg		String	JWS alg algorithm (JWA) that must be used for signing the JWT used to authenticate the Client at the Token Endpoint for the private_key_jwt and client_secret_jwt authentication methods.	
default_max_age		Integer	Specifies the Default Maximum Authentication Age.	

Field Name	Required	Type	Description	Format
require_auth_time		Boolean	Boolean value specifying whether the auth_time Claim in the ID Token is required. It is required when the value is true.	
default_acr_values		List of [string]	Array of default requested Authentication Context Class Reference values that the Authorization Server must use for processing requests from the Client.	
initiate_login_uri		String	Specifies the URI using the https scheme that the authorization server can call to initiate a login at the client.	
post_logout_redirect_uris		List of [string]	Provide the URLs supplied by the RP to request that the user be redirected to this location after a logout has been performed.	
frontchannel_logout_uri		String	RP URL that will cause the RP to log itself out when rendered in an iframe by the OP.	
frontchannel_logout_session_required		Boolean	Boolean value specifying whether the RP requires that a session ID query parameter be included to identify the RP session at the OP when the logout_uri is used. If omitted, the default value is false.	
backchannel_logout_uri		String	RP URL that will cause the RP to log itself out when sent a Logout Token by the OP.	
backchannel_logout_session_required		Boolean	Boolean value specifying whether the RP requires that a session ID Claim be included in the Logout Token to identify the RP session with the OP when the backchannel_logout_uri is used. If omitted, the default value is false.	
request_uris		List of [string]	Provide a list of request_uri values that are pre-registered by the Client for use at the Authorization Server.	
scopes		String	This param will be removed in a future version because the correct is 'scope' not 'scopes', see (rfc7591).	

Field Name	Required	Type	Description	Format
claims		String	String containing a space-separated list of claims that can be requested individually.	
id_token_token_binding_cnf		String	Specifies the JWT Confirmation Method member name (e.g. tbh) that the Relying Party expects when receiving Token Bound ID Tokens. The presence of this parameter indicates that the Relying Party supports Token Binding of ID Tokens. If omitted, the default is that the Relying Party does not support Token Binding of ID Tokens.	
tls_client_auth_subject_dn		String	An string representation of the expected subject distinguished name of the certificate, which the OAuth client will use in mutual TLS authentication.	
allow_spontaneous_scopes		Boolean	Specifies whether to allow spontaneous scopes for client. The default value is false.	
spontaneous_scopes		List of [string]	List of spontaneous scopes	
run_introspection_script_before_access_token_as_jwt_creation_and_include_claims		Boolean	Boolean value with default value false. If true and access_token_as_jwt\=true then run introspection script and transfer claims into JWT.	
keep_client_authorization_after_expiration		Boolean	Boolean value indicating if the client authorization will not be removed afer expiration (expiration date is same as client's expiration that created it). The default value is false.	
scope		List of [string]	Provide list of scope which are used during authentication to authorize access to resource.	
authorized_origins		List of [string]	specifies authorized JavaScript origins.	
access_token_lifetime		Integer	Specifies the Client-specific access token expiration.	

Field Name	Required	Type	Description	Format
software_id		String	Specifies a unique identifier string (UUID) assigned by the client developer or software publisher used by registration endpoints to identify the client software to be dynamically registered.	
software_version		String	Specifies a version identifier string for the client software identified by 'software_id'. The value of the 'software_version' should change on any update to the client software identified by the same 'software_id'.	
software_statement		String	specifies a software statement containing client metadata values about the client software as claims. This is a string value containing the entire signed JWT.	
backchannel_token_delivery_mode		String	specifies how backchannel token will be deliveried.	
backchannel_client_notification_endpoint		String	Client Initiated Backchannel Authentication (CIBA) enables a Client to initiate the authentication of an end-user by means of out-of-band mechanisms. Upon receipt of the notification, the Client makes a request to the token endpoint to obtain the tokens.	
backchannel_authentication_request_signing_alg		String	The JWS algorithm alg value that the Client will use for signing authentication request, as described in Section 7.1.1. of OAuth 2.0 [RFC6749]. When omitted, the Client will not send signed authentication requests.	
backchannel_user_code_parameter		Boolean	Boolean value specifying whether the Client supports the user_code parameter. If omitted, the default value is false.	
additional_audience		List of [string]	Additional audiences.	

3.2.24. *RegisterResponseParam*

Field Name	Required	Type	Description	Format
client_id	X	String	Unique Client Identifier. It MUST NOT be currently valid for any other registered Client.	
client_secret		String	This value is used by Confidential Clients to authenticate to the Token Endpoint	
registration_access_token		String	Registration Access Token that can be used at the Client Configuration Endpoint to perform subsequent operations upon the Client registration.	
registration_client_uri		String	Location of the Client Configuration Endpoint where the Registration Access Token can be used to perform subsequent operations upon the resulting Client registration.	
client_id_issued_at		Integer	Time at which the Client Identifier was issued.	
client_secret_expires_at		Integer	Time at which the client_secret will expire or 0 if it will not expire.	

3.2.25. *RptIntrospectionResponse*

Field Name	Required	Type	Description	Format
active	X	Boolean	Boolean indicator of whether or not the presented token is currently active.	
exp		Long	Integer timestamp, in seconds since January 1 1970 UTC, indicating when this token will expire.	int64
iat		Integer	Integer timestamp, measured in the number of seconds since January 1 1970 UTC, indicating when this permission was originally issued.	
clientId		String	Client id used to obtain RPT.	

Field Name	Required	Type	Description	Format
sub		String	Subject of the token. Usually a machine-readable identifier of the resource owner who authorized this token.	
aud		String	Service-specific string identifier or list of string identifiers representing the intended audience for this token.	
permissions	X	List of [RptIntrospectionResponse_permissions]		
pct_claims		Map of [string]	PCT token claims.	
iss		String	String representing the issuer of this token, as defined in JWT [RFC7519].	
jti		String	String identifier for the token, as defined in JWT [RFC7519].	
nbf		Integer	Integer timestamp, measured in the number of seconds since January 1 1970 UTC, indicating the time before which this permission is not valid.	
resource_id	X	String	Resource ID.	
resource_scopes	X	List of [string]		

3.2.26. *RptIntrospectionResponse1*

Field Name	Required	Type	Description	Format
active	X	Boolean	Boolean indicator of whether or not the presented token is currently active.	
exp		Long	Integer timestamp, in seconds since January 1 1970 UTC, indicating when this token will expire.	int64
iat		Integer	Integer timestamp, measured in the number of seconds since January 1 1970 UTC, indicating when this permission was originally issued.	
clientId		String	Client id used to obtain RPT.	

Field Name	Required	Type	Description	Format
sub		String	Subject of the token. Usually a machine-readable identifier of the resource owner who authorized this token.	
aud		String	Service-specific string identifier or list of string identifiers representing the intended audience for this token.	
permissions	X	List of [RptIntrospectionResponse_permissions]		
pct_claims		Map of [string]		
iss		String	String representing the issuer of this token, as defined in JWT [RFC7519].	
jti		String	String identifier for the token, as defined in JWT [RFC7519].	
nbf		Integer	Integer timestamp, measured in the number of seconds since January 1 1970 UTC, indicating the time before which this permission is not valid.	
resource_id	X	String	Resource ID.	
resource_scopes	X	List of [string]		

3.2.27. *RptIntrospectionResponsePermissions*

List of UmaPermission granted to RPT. A permission is (requested or granted) authorized access to a particular resource with some number of scopes bound to that resource.

Field Name	Required	Type	Description	Format
resource_id	X	String	A string that uniquely identifies the protected resource, access to which has been granted to this client on behalf of this requesting party. The identifier MUST correspond to a resource that was previously registered as protected.	

Field Name	Required	Type	Description	Format
resource_scopes	X	List of [string]	An array referencing zero or more strings representing scopes to which access was granted for this resource. Each string MUST correspond to a scope that was registered by this resource server for the referenced resource.	
exp		Integer	Integer timestamp, measured in the number of seconds since January 1 1970 UTC, indicating when this permission will expire. If the token-level exp value pre-dates a permission-level exp value, the token-level value takes precedence.	
iat		Integer	Integer timestamp, measured in the number of seconds since January 1 1970 UTC, indicating when this permission was originally issued. If the token-level iat value post-dates a permission-level iat value, the token-level value takes precedence.	
nbf		Integer	Integer timestamp, measured in the number of seconds since January 1 1970 UTC, indicating the time before which this permission is not valid. If the token-level nbf value post-dates a permission-level nbf value, the token-level value takes precedence.	

3.2.28. *SessionStateObject*

Field Name	Required	Type	Description	Format
state		String	String that represents the End-User's login state at the OP. It MUST NOT contain the space (\\" \\") character.	
auth_time		date	specifies the time at which session was authenticated.	date
custom_state		String		

3.2.29. *UmaPermissiona* UmaPermissiona

A permission is (requested or granted) authorized access to a particular resource with some number of scopes bound to that resource.

Field Name	Required	Type	Description	Format
resource_id	X	String	The identifier for a resource to which this client is seeking access. The identifier MUST correspond to a resource that was previously registered.	
resource_scopes	X	List of [string]	An array referencing zero or more strings representing scopes to which access was granted for this resource. Each string MUST correspond to a scope that was registered by this resource server for the referenced resource.	
params		Map of [string]	A key/value map that can contain custom parameters.	

3.2.30. *UmaResource* UmaResource

Resource description

Field Name	Required	Type	Description	Format
name		String	A human-readable string describing a set of one or more resources. This name MAY be used by the authorization server in its resource owner user interface for the resource owner.	
icon_uri		String	A URI for a graphic icon representing the resource set. The referenced icon MAY be used by the authorization server in its resource owner user interface for the resource owner.	

Field Name	Required	Type	Description	Format
type		String	A string uniquely identifying the semantics of the resource set. For example, if the resource set consists of a single resource that is an identity claim that leverages standardized claim semantics for \\"verified email address\\", the value of this property could be an identifying URI for this claim.	
resource_scopes	X	List of [string]	An array of strings, any of which MAY be a URI, indicating the available scopes for this resource set. URIs MUST resolve to scope descriptions as defined in Section 2.1. Published scope descriptions MAY reside anywhere on the web; a resource server is not required to self-host scope descriptions and may wish to point to standardized scope descriptions residing elsewhere. It is the resource server's responsibility to ensure that scope description documents are accessible to authorization servers through GET calls to support any user interface requirements. The resource server and authorization server are presumed to have separately negotiated any required interpretation of scope handling not conveyed through scope descriptions.	
scope_expression		String		
description		String	A human-readable string describing the resource	
iat		Long	number of seconds since January 1 1970 UTC, indicating when the token was issued at	int64
exp		Long	number of seconds since January 1 1970 UTC, indicating when this token will expire.	int64

3.2.31. *UmaResource1* UmaResource

Resource description

Field Name	Required	Type	Description	Format
name		String	A human-readable string describing a set of one or more resources. This name MAY be used by the authorization server in its resource owner user interface for the resource owner.	
icon_uri		String	A URI for a graphic icon representing the resource set. The referenced icon MAY be used by the authorization server in its resource owner user interface for the resource owner.	
type		String	A string uniquely identifying the semantics of the resource set. For example, if the resource set consists of a single resource that is an identity claim that leverages standardized claim semantics for \\"verified email address\\", the value of this property could be an identifying URI for this claim.	
resource_scopes	X	List of [string]	An array of strings, any of which MAY be a URI, indicating the available scopes for this resource set. URIs MUST resolve to scope descriptions as defined in Section 2.1. Published scope descriptions MAY reside anywhere on the web; a resource server is not required to self-host scope descriptions and may wish to point to standardized scope descriptions residing elsewhere. It is the resource server's responsibility to ensure that scope description documents are accessible to authorization servers through GET calls to support any user interface requirements. The resource server and authorization server are presumed to have separately negotiated any required interpretation of scope handling not conveyed through scope descriptions.	
scope_expression		String		

Field Name	Required	Type	Description	Format
description		String	A human-readable string describing the resource	
iat		Long	number of seconds since January 1 1970 UTC, indicating when the token was issued at	int64
exp		Long	number of seconds since January 1 1970 UTC, indicating when this token will expire.	int64

3.2.32. *UmaResourceResponse*

UmaResourceResponse Resource created.

Field Name	Required	Type	Description	Format
_id	X	String	UMA Resource identifier	
user_access_policy_uri		String		

3.2.33. *UmaResourceWithId*

Uma Resource details

Field Name	Required	Type	Description	Format
_id	X	String	UMA Resource identifier	
name		String	A human-readable string describing a set of one or more resources. This name MAY be used by the authorization server in its resource owner user interface for the resource owner.	
uri		String	A human-readable string describing the resource	

Field Name	Required	Type	Description	Format
type		String	A string uniquely identifying the semantics of the resource set. For example, if the resource set consists of a single resource that is an identity claim that leverages standardized claim semantics for \\"verified email address\\", the value of this property could be an identifying URI for this claim.	
scopes		List of [string]	An array of strings, any of which MAY be a URI, indicating the available scopes for this resource set. URIs MUST resolve to scope descriptions as defined in Section 2.1. Published scope descriptions MAY reside anywhere on the web; a resource server is not required to self-host scope descriptions and may wish to point to standardized scope descriptions residing elsewhere. It is the resource server's responsibility to ensure that scope description documents are accessible to authorization servers through GET calls to support any user interface requirements. The resource server and authorization server are presumed to have separately negotiated any required interpretation of scope handling not conveyed through scope descriptions.	
scope_expression		String		
description		String	A human-readable string describing the resource	
icon_uri		String	A URI for a graphic icon representing the resource set. The referenced icon MAY be used by the authorization server in its resource owner user interface for the resource owner.	
iat	X	Long	number of seconds since January 1 1970 UTC, indicating when the token was issued at	int64

Field Name	Required	Type	Description	Format
exp	X	Long	number of seconds since January 1 1970 UTC, indicating when this token will expire.	int64

3.2.34. *WebKeysConfiguration*

JSON Web Key Set (JWKS) - A JSON object that represents a set of JWKs. The JSON object MUST have a keys member, which is an array of JWKs.

Field Name	Required	Type	Description	Format
keys	X	List of *JsonWebKey*	List of JSON Web Key (JWK) - A JSON object that represents a cryptographic key. The members of the object represent properties of the key, including its value.	

3.2.35. *CustomUserAttributes*

Defines custom user attributes that are used to expand the default gluuPerson objectclass, by using the gluuCustomPerson objectclass

Field Name	Required	Type	Description	Format
isOperator	-	Boolean	Flag that signals if a registered user is an operator.	

<< End of Document >>