International Standard

ISO/IEC 26138

First edition
2024-10

# Information technology — OpenID connect — OAuth 2.0 multiple response type encoding practices



Reference number
ISO/IEC 26138:2024(en)

**COPYRIGHT PROTECTED DOCUMENT**



ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

# Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and https://patents.iec.ch. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by the OpenID Foundation (OIDF) (as OAuth 2.0 Multiple Response Type Encoding Practices) and drafted in accordance with its editorial rules. It was adopted, under the JTC 1 PAS procedure, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

**Abstract**

This specification provides guidance on the proper encoding of responses to OAuth 2.0 Authorization Requests in which the request uses a Response Type value that includes space characters. Furthermore, this specification registers several new Response Type values in the OAuth Authorization Endpoint Response Types registry.

This specification also defines a Response Mode Authorization Request parameter that informs the Authorization Server of the mechanism to be used for returning Authorization Response parameters from the Authorization Endpoint.

Table of Contents

1. Introduction
   1.1. Requirements Notation and Conventions
   1.2. Terminology
2. Response Types and Response Modes
   2.1. Response Modes
   2.2. Multiple-Valued Response Types
3. ID Token Response Type
4. None Response Type
5. Definitions of Multiple-Valued Response Type Combinations
6. IANA Considerations
   6.1. OAuth Authorization Endpoint Response Types Registration
      6.1.1. Registry Contents
   6.2. OAuth Parameters Registration
      6.2.1. Registry Contents
7. Security Considerations
8. References
   8.1. Normative References
   8.2. Informative References
Appendix A. Example using Multiple-Valued Response Type

# Information technology — OpenID Connect — OAuth 2.0 Multiple Response Type Encoding Practices

## 1. Introduction

TOC

---

### 1.1. Requirements Notation and Conventions

TOC

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

In the .txt version of this document, values are quoted to indicate that they are to be taken literally. When using these values in protocol messages, the quotes MUST NOT be used as part of the value. In the HTML version of this document, values to be taken literally are indicated by the use of `this fixed-width font`.

---

### 1.2. Terminology

TOC

This specification uses the terms "Access Token", "Authorization Code", "Authorization Endpoint", "Authorization Grant", "Authorization Server", "Client", "Client Identifier", "Client Secret", "Protected Resource", "Redirection URI", "Refresh Token", "Resource Owner", "Resource Server", "Response Type", and "Token Endpoint" defined by OAuth 2.0 [RFC6749] and the term "User Agent" defined by RFC 2616 [RFC2616]. This specification also defines the following terms:

Multiple-Valued Response Types

> The OAuth 2.0 specification allows for registration of space-separated `response_type` parameter values. If a Response Type contains one of more space characters (%20), it is compared as a space-delimited list of values in which the order of values does not matter.

Response Mode

The Response Mode determines how the Authorization Server returns result parameters from the Authorization Endpoint. Non-default modes are specified using the `response_mode` request parameter. If `response_mode` is not present in a request, the default Response Mode mechanism specified by the Response Type is used.

---

## 2. Response Types and Response Modes

TOC

The Response Type request parameter `response_type` informs the Authorization Server of the desired authorization processing flow, including what parameters are returned from the endpoints used. The Response Mode request parameter `response_mode` informs the Authorization Server of the mechanism to be used for returning Authorization Response parameters from the Authorization Endpoint. Each Response Type value also defines a default Response Mode mechanism to be used, if no Response Mode is specified using the request parameter.

---

### 2.1. Response Modes

TOC

This specification defines the following OAuth Authorization Request parameter:

response_mode

OPTIONAL. Informs the Authorization Server of the mechanism to be used for returning Authorization Response parameters from the Authorization Endpoint. This use of this parameter is NOT RECOMMENDED with a value that specifies the same Response Mode as the default Response Mode for the Response Type used.

This specification defines the following Response Modes, which are described with their `response_mode` parameter values:

query

> In this mode, Authorization Response parameters are encoded in the query string added to the `redirect_uri` when redirecting back to the Client.

fragment

> In this mode, Authorization Response parameters are encoded in the fragment added to the `redirect_uri` when redirecting back to the Client.

For purposes of this specification, the default Response Mode for the OAuth 2.0 `code` Response Type is the `query` encoding. For purposes of this specification, the default Response Mode for the OAuth 2.0 `token` Response Type is the `fragment` encoding.

See OAuth 2.0 Form Post Response Mode [OAuth.Post] for an example of a specification that defines an additional Response Mode. Note that it is expected that additional Response Modes may be defined by other specifications in the future, including possibly ones utilizing the HTML5 postMessage API and Cross Origin Resource Sharing (CORS).

---

### 2.2. Multiple-Valued Response Types

TOC

When a multiple-valued Response Type is defined, it is RECOMMENDED that the following encoding rules be applied for the issued response from the Authorization Endpoint.

All parameters returned from the Authorization Endpoint SHOULD use the same Response Mode. This recommendation applies to both success and error responses.

Rationale: This significantly simplifies Client parameter processing. It also can have positive performance benefits, as described below.

For instance, if a response includes fragment encoded parts, a User Agent Client component must be involved to complete processing of the response. If a new query parameter is added to the Client URI, it will cause the User Agent to re-fetch the Client URI, causing discontinuity of operation of the User Agent based Client components. If only fragment encoding is used, the User Agent will simply reactivate the Client component, which can then process the fragment and also convey any

parameters to a Client host as necessary, e.g., via XmlHttpRequest. Therefore, full fragment encoding always results in lower latency for response processing.

---

## 3. ID Token Response Type

TOC

This section registers a new Response Type, the `id_token`, in accordance with the stipulations in the OAuth 2.0 specification, Section 8.4. The intended purpose of the `id_token` is that it MUST provide an assertion of the identity of the Resource Owner as understood by the Authorization Server. The assertion MUST specify a targeted audience, e.g. the requesting Client. However, the specific semantics of the assertion and how it can be validated are not specified in this document.

id_token

> When supplied as the `response_type` parameter in an OAuth 2.0 Authorization Request, a successful response MUST include the parameter `id_token`. The Authorization Server SHOULD NOT return an OAuth 2.0 Authorization Code, Access Token, or Access Token Type in a successful response to the grant request. If a `redirect_uri` is supplied, the User Agent SHOULD be redirected there after granting or denying access. The request MAY include a `state` parameter, and if so, the Authorization Server MUST echo its value as a response parameter when issuing either a successful response or an error response. The default Response Mode for this Response Type is the fragment encoding and the query encoding MUST NOT be used. Both successful and error responses SHOULD be returned using the supplied Response Mode, or if none is supplied, using the default Response Mode.

Returning the `id_token` in a fragment reduces the likelihood that the `id_token` leaks during transport and mitigates the associated risks to the privacy of the user (Resource Owner).

## 4. None Response Type

TOC

This section registers the Response Type `none`, in accordance with the stipulations in the OAuth 2.0 specification, Section 8.4. The intended purpose is to enable use cases where a party requests the Authorization Server to register a grant of access to a Protected Resource on behalf of a Client but requires no access credentials to be returned to the Client at that time. The means by which the Client eventually obtains the access credentials is left unspecified here.

One scenario is where a user wishes to purchase an application from a market, and desires to authorize application installation and grant the application access to Protected Resources in a single step. However, since the user is not presently interacting with the (not yet active) application, it is not appropriate to return access credentials simultaneously in the authorization step.

none

> When supplied as the `response_type` parameter in an OAuth 2.0 Authorization Request, the Authorization Server SHOULD NOT return an OAuth 2.0 Authorization Code, Access Token, Access Token Type, or ID Token in a successful response to the grant request. If a `redirect_uri` is supplied, the User Agent SHOULD be redirected there after granting or denying access. The request MAY include a `state` parameter, and if so, the Authorization Server MUST echo its value as a response parameter when issuing either a successful response or an error response. The default Response Mode for this Response Type is the query encoding. Both successful and error responses SHOULD be returned using the supplied Response Mode, or if none is supplied, using the default Response Mode.

The Response Type `none` SHOULD NOT be combined with other Response Types.

## 5. Definitions of Multiple-Valued Response Type Combinations

TOC

This section defines combinations of the values `code`, `token`, and `id_token`, which are each individually registered Response Types.

code token

When supplied as the value for the `response_type` parameter, a successful response MUST include an Access Token, an Access Token Type, and an Authorization Code. The default Response Mode for this Response Type is the fragment encoding and the query encoding MUST NOT be used. Both successful and error responses SHOULD be returned using the supplied Response Mode, or if none is supplied, using the default Response Mode.

code id_token

When supplied as the value for the `response_type` parameter, a successful response MUST include both an Authorization Code and an `id_token`. The default Response Mode for this Response Type is the fragment encoding and the query encoding MUST NOT be used. Both successful and error responses SHOULD be returned using the supplied Response Mode, or if none is supplied, using the default Response Mode.

id_token token

When supplied as the value for the `response_type` parameter, a successful response MUST include an Access Token, an Access Token Type, and an `id_token`. The default Response Mode for this Response Type is the fragment encoding and the query encoding MUST NOT be used. Both successful and error responses SHOULD be returned using the supplied Response Mode, or if none is supplied, using the default Response Mode.

code id_token token

When supplied as the value for the `response_type` parameter, a successful response MUST include an Authorization Code, an `id_token`, an Access Token, and an Access Token Type. The default Response Mode for this Response Type is the fragment encoding and the query encoding MUST NOT be used. Both successful and error responses SHOULD be returned using the supplied Response

Mode, or if none is supplied, using the default Response Mode.

For all these Response Types, the request MAY include a `state` parameter, and if so, the Authorization Server MUST echo its value as a response parameter when issuing either a successful response or an error response.

A non-normative request/response example as issued/received by the User Agent (with extra line breaks for display purposes only) is:

```
  GET /authorize?
    response_type=id_token%20token
    &client_id=s6BhdRkqt3
    &redirect_uri=https%3A%2F%2Fclient.example.org%2Fcb
    &state=af0ifjsldkj HTTP/1.1
  Host: server.example.com

  HTTP/1.1 302 Found
  Location: https://client.example.org/cb#
  access_token=SlAV32hkKG
  &token_type=bearer
  &id_token=eyJ0 ... NiJ9.eyJ1c ... I6IjIifX0.DeWt4Qu
... ZXso
  &expires_in=3600
  &state=af0ifjsldkj
```



## 6. IANA Considerations

TOC

### 6.1. OAuth Authorization Endpoint Response Types Registration

TOC

This specification registers the `response_type` values defined by this specification in the IANA OAuth Authorization Endpoint Response Types registry defined in RFC 6749 [RFC6749].

### 6.1.1. Registry Contents

TOC

- Response Type Name: `id_token`
- Change Controller: OpenID Foundation Artifact Binding Working Group - openid-specs-ab@lists.openid.net
- Specification Document(s): http://openid.net/specs/oauth-v2-multiple-response-types-1_0.html
- Response Type Name: `none`
- Change Controller: OpenID Foundation Artifact Binding Working Group - openid-specs-ab@lists.openid.net
- Specification Document(s): http://openid.net/specs/oauth-v2-multiple-response-types-1_0.html
- Response Type Name: `code token`
- Change Controller: OpenID Foundation Artifact Binding Working Group - openid-specs-ab@lists.openid.net
- Specification Document(s): http://openid.net/specs/oauth-v2-multiple-response-types-1_0.html
- Response Type Name: `code id_token`
- Change Controller: OpenID Foundation Artifact Binding Working Group - openid-specs-ab@lists.openid.net
- Specification Document(s): http://openid.net/specs/oauth-v2-multiple-response-types-1_0.html
- Response Type Name: `id_token token`
- Change Controller: OpenID Foundation Artifact Binding Working Group - openid-specs-ab@lists.openid.net
- Specification Document(s): http://openid.net/specs/oauth-v2-multiple-response-types-1_0.html
- Response Type Name: `code id_token token`
- Change Controller: OpenID Foundation Artifact Binding Working Group - openid-specs-ab@lists.openid.net