
# Policy Decision Point Interface Control Document

***EOEPCA.ICD.xxx***

TVUK System Team

Version 1.0, 30/11/2020:

# Policy Decision Point Interface Control Document

EO Exploitation Platform Common Architecture
*Policy Decision Point 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-pdp-engine/issues. | This document is available in PDF format here. |
| **EUROPEAN SPACE AGENCY CONTRACT REPORT**<br>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**<br>350 Capability Green, Luton, Bedfordshire, LU1 3LU, United Kingdom.<br>Tel: +44 (0)1582 399000<br>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 Policy Decision Point 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 Policy Decision Point. The ICD provides a Building Block level specification of the interfaces exposed by the PDP to the rest of EOEPCA components.

**Section [Interfaces]**

Provides the interface specification of the Building Block.

# Chapter 3. Policy Decision Point Interfaces


### Abstract

***This OpenAPI Document describes the endpoints exposed by Policy Decision*** *Point Building Block deployments. <br> <br> Using this API will allow to register policies that support the protection of policies using both the Login Service and the Policy Decision Point and to perform checks based on XACML requests and responses*


## 3.1. Endpoints

### 3.1.1. Policy Checks

#### 3.1.1.1. Policy Validation

`POST /policy/validate`

XACML Policy Checks

##### 3.1.1.1.1. Description

This operation lists all policies filtered by ownership ID. Ownership ID is extracted from the OpenID Connect Token

##### 3.1.1.1.2. Parameters

##### 3.1.1.1.3. Return Type

-

##### 3.1.1.1.4. Responses

*Table 2. http response codes*

| Code | Message | Datatype |
|---|---|---|
| 200 | OK | <<>> |

##### 3.1.1.1.5. Samples

### 3.1.2. Policy Management

#### 3.1.2.1. Policy GET

`GET /policy`

List all owned policies

#### 3.1.2.1.1. Description

This operation lists all policies filtered by ownership ID. Ownership ID is extracted from the OpenID Connect Token

#### 3.1.2.1.2. Parameters

### Header Parameters

| Name | Description | Required |
|---|---|---|
| Authorization | JWT or Bearer Token | - |

#### 3.1.2.1.3. Return Type

array[[policy]]

#### 3.1.2.1.4. Content Type

- application/json

#### 3.1.2.1.5. Responses

*Table 3. http response codes*

| Code | Message | Datatype |
|---|---|---|
| 200 | OK | List[[policy]] |

#### 3.1.2.1.6. Samples

### 3.1.2.2. Policy DELETE (ID)

`DELETE /policy/{policy_id}`

Deletes an owned Policy Reference from the Platform

#### 3.1.2.2.1. Description

This operation removes an existing Policy reference owned by the user.

#### 3.1.2.2.2. Parameters

### Path Parameters

| Name | Description | Required |
|---|---|---|
| policy_id | Unique Policy ID | X |

### Header Parameters

| Name | Description | Required |
|---|---|---|
| Authorization | JWT or Bearer Token | - |

#### 3.1.2.2.3. Return Type

-

#### 3.1.2.2.4. Responses

*Table 4. http response codes*

| Code | Message | Datatype |
|---|---|---|
| 200 | OK | <<>> |
| 401 | UNAUTHORIZED | <<>> |
| 404 | NOT FOUND | <<>> |

#### 3.1.2.2.5. Samples

### 3.1.2.3. Policy GET (ID)

`GET /policy/{policy_id}`

Retrieve a specific owned policy

#### 3.1.2.3.1. Description

This operation retrieves information about an owned policy.

#### 3.1.2.3.2. Parameters

#### Path Parameters

| Name | Description | Required |
|---|---|---|
| policy_id | Unique Policy ID | X |

#### Header Parameters

| Name | Description | Required |
|---|---|---|
| Authorization | JWT or Bearer Token | - |

#### 3.1.2.3.3. Return Type

[policy]

#### 3.1.2.3.4. Content Type

- application/json

#### 3.1.2.3.5. Responses

*Table 5. http response codes*

| Code | Message | Datatype |
|---|---|---|
| 200 | OK | [policy] |
| 404 | NOT FOUND | <<>> |

#### 3.1.2.3.6. Samples

### 3.1.2.4. Policy PUT (ID)

`PUT /policy/{policy_id}`

Updates an existing Policy reference in the Platform

#### 3.1.2.4.1. Description

This operation updates an existing 'owned' policy reference.

#### 3.1.2.4.2. Parameters

### Path Parameters

| Name | Description | Required |
|---|---|---|
| policy_id | Unique Policy ID | X |

#### 3.1.2.4.3. Body Parameter

| Name | Description | Required |
|---|---|---|
| Policy | *Policy* | X |

### Header Parameters

| Name | Description | Required |
|---|---|---|
| Authorization | JWT or Bearer Token | - |

#### 3.1.2.4.4. Return Type

-

#### 3.1.2.4.5. Responses

*Table 6. http response codes*

| Code | Message | Datatype |
|---|---|---|
| 200 | OK | <<>> |
| 401 | UNAUTHORIZED | <<>> |
| 404 | NOT FOUND | <<>> |

#### 3.1.2.4.6. Samples

#### 3.1.2.5. Policy POST

`POST /policy`

Creates a new Policy reference in the Platform

##### 3.1.2.5.1. Description

This operation generates a new policy reference object that can be protected. Ownership ID is set to the unique ID of the End-User

##### 3.1.2.5.2. Parameters

##### 3.1.2.5.3. Body Parameter

| Name | Description | Required |
|---|---|---|
| NewPolicy | *NewPolicy* | X |

**Header Parameters**

| Name | Description | Required |
|---|---|---|
| Authorization | JWT or Bearer Token | - |

##### 3.1.2.5.4. Return Type

[policy]

##### 3.1.2.5.5. Content Type

- application/json

##### 3.1.2.5.6. Responses

*Table 7. http response codes*

| Code | Message | Datatype |
|---|---|---|
| 200 | OK | [policy] |
| 401 | UNAUTHORIZED | <<>> |
| 404 | NOT FOUND | <<>> |

##### 3.1.2.5.7. Samples

### 3.1.3. API

#### 3.1.3.1. Swagger UI

`/swagger-ui`

#### 3.1.3.1.1. Description

This operation accesses the API for the Policy Decision Point

#### 3.1.3.1.2. Parameters

**Path Parameters**

| Name | Description | Required |
|---|---|---|
| - | - | - |

**Header Parameters**

| Name | Description | Required |
|---|---|---|
| - | - | - |

#### 3.1.3.1.3. Return Type

-

#### 3.1.3.1.4. Responses

*Table 8. http response codes*

| Code | Message | Datatype |
|---|---|---|
| 200 | OK | <<>> |

#### 3.1.3.1.5. Samples

# 3.2. Models

## 3.2.1. *NewPolicy*

| Field Name | Required | Type | Description | Format |
|---|---|---|---|---|
| name | | String | Human readable name for the policy | |
| description | | String | Custom description for the policy pourpose | |
| policy_cfg | | policy_cfg | | |
| scopes | | List of [string] | List of scopes associated with the policy | |

## 3.2.2. *Policy*

| Field Name | Required | Type | Description | Format |
|---|---|---|---|---|
| ownership_id | | UUID | UUID of the Owner End-User | uuid |
| id | | UUID | UUID of the policy | uuid |
| name | | String | Human readable name for the policy | |
| description | | String | Custom description for the policy pourpose | |
| policy_cfg | | policy_cfg | | |
| scopes | | List of [string] | List of scopes associated with the policy | |

### 3.2.3. *PolicyCfg*

Custom rules for the policy configuration

| Field Name | Required | Type | Description | Format |
|---|---|---|---|---|
| resource_id | | UUID | UUID of the applicable Resource | uuid |
| action | | String | Applicable XACML Action | |
| rules | | List of [object] | List of checks to create a policy. Each one will be executed in sequence, and the policy will fail as soon as one fails | |

<< End of Document >>