

CICD-OIC3-quickstart

This is CICD starter Oracle Integration Cloud(OIC) REST APIs collection. This collection contains a few of the use cases and a ready-made collection of requests that can be used.

This collection uses **variables that are defined in an collection.**

For more see [OIC REST API documentation](#)

AuthorizationOAuth 2.0

AccessToken

As the prerequisite you must create in the [IAM Domain Integrated application](#) with the checked **Grant Type : Client Credentials or Authorization Code**

Then the requests provides for you the way to get the token which works with OIC REST management API and that token should be used to process other folders requests:

Grant Type : Client Credentials / Authorization Code Access Token URL : https://[idcs url]/oauth2/v1/token **Client ID:** xxxxxx **Client Secret:** xxxxxx **Scope:** https://[ocp-resource].oraclecloud.com:443/ic/api/

Note: https://[ocp-resource].oraclecloud.com:443urn:opc:resource:consumer::all - is scope for the deployed integrations invoke

Optional for the postman token retrieval configuration:

Auth URL* : https://[idcs url]/oauth2/v1/authorize Redirect URL* : https://[oic url]/icsapis/agent/oauth/callback

AuthorizationOAuth 2.0

This folder is using an authorization helper from collection CICD-OIC3-quickstart

POSTOIC3 with OAuth Client Credentials

[Open request](#)

/oauth2/v1/token

Add request description...

AuthorizationBasic Auth

Username

<<your-IntegratedApp-ClientID>>

Password

<<your-IntegratedApp-Secret>>

Bodyurlencoded

grant_type

client_credentials

scope

https://<<your-serviceResource-scope>>.integration.<<your-OIC-Instance-region>>.ocp.oraclecloud.com:443/ic/api/

Example

Obtain access_token (client credentials) - OIC

Request

cURL

```
curl --location '/oauth2/v1/token' \
--data-urlencode 'grant_type=client_credentials' \
--data-urlencode 'scope={{OIC_SCOPE}}'
200 OK
```

Response

- Body
- Headers (14)

View More

```
json
{
```

POSTOIC3 DR Instance with OAuth Client Credentials

/oauth2/v1/token

AuthorizationBasic Auth

<<your-IntegratedApp-ClientID>>

<<your-IntegratedApp-Secret>>

```
client_credentials
```

Example

Obtain access_token (client credentials) - OIC

Request

cURL

```
curl --location '/oauth2/v1/token' \
--data-urlencode 'grant_type=client_credentials' \
--data-urlencode 'scope={{OIC_SCOPE}}'
200 OK
```

Response

- Body
- Headers (14)

[View More](#)

json

[illegible]

Connections

This folder contains requests that allow to [get connection details and use it for the update connections](#) in the global Design menu of OIC

Typical pipeline between the environments is to get(export) connection details to file(JSON payload), modify the connection details according target environment and after import the integration follows update connection operation.

AuthorizationOAuth 2.0

This folder is using an authorization helper from collection CICD-OIC3-quickstart

GETOIC3_DEV_connections

[Open request](#)

https://design.integration.<<your-OIC-Instance-region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections?integrationInstance=<<your-OIC-InstanceName>>

Add request description...

AuthorizationOAuth 2.0

Query Params

integrationInstance

<<your-OIC-InstanceName>>

GETOIC3_DEV_connectionTOTEST

[Open request](#)

https://design.integration.<<your-OIC-Instance-region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/?integrationInstance=<<your-OIC-InstanceName>>

Add request description...

AuthorizationOAuth 2.0

Query Params

integrationInstance

<<your-OIC-InstanceName>>

Example

OIC3_DEV_connectionTOTEST

Request

View More

cURL

```
curl --location 'https://design.integration.<<your-OIC-Instance-region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/?integrationInstance=%3C%3Cyour-OIC-InstanceName%3E%3E'  
200 OK
```

Response

- Body
- Headers (12)

View More

json

```
{  
  "adapterType": {  
    "displayName": "REST",  
    "links": [],  
    "name": "rest",  
    "type": "PREINSTALLED"  
  },  
  "agentRequired": false,  
  "agentSupported": true,  
  "connectionProperties": [  
    {  
      "acceptableKeys": [  
        {  
          "displayName": "REST API Base URL",  
          "keyName": "restUrl",  
          "scope": "SOURCE_AND_TARGET"  
        },  
        {  
          "displayName": "Open API (1.0/2.0/3.0) URL",  
          "keyName": "openApiUrl",  
          "scope": "SOURCE_AND_TARGET"  
        }  
      ],  
      "displayName": "Connection Type",  
      "hasAttachment": false,  
    }  
  ]  
}
```

```

        "hiddenFlag": false,
        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "connectionType",
        "propertyType": "CHOICE",
        "propertyValue": "restUrl",
        "requiredFlag": true
    },
    {
        "acceptableKeys": [
            {
                "displayName": "TLSv1.1",
                "keyName": "TLSv1.1",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "TLSv1.2",
                "keyName": "TLSv1.2",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "displayName": "TLS Version",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "tlsVersion",
        "propertyShortDesc": "TLS Version",
        "propertyType": "CHOICE",
        "requiredFlag": false
    },
    {
        "displayName": "Connection URL",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "connectionUrl",
        "propertyShortDesc": "Please make sure that this value really corresponds to the type selected above.",
        "propertyType": "URL",
        "propertyValue": "https://opa-oic-teamoic3-1740047240850-<<Storage-namespace>>-fra.process.oci.oraclecloud.com/process/api",
        "requiredFlag": true
    },
    {
        "acceptableKeys": [
            {

```

```

        "displayName": "Yes",
        "keyName": "true",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "No",
        "keyName": "false",
        "scope": "SOURCE_AND_TARGET"
    }
],
"displayName": "Enable two way SSL for outbound connections (Optional)",
"hasAttachment": false,
"hiddenFlag": false,
"propertyDescription": "Both the client and the server pass certificates to each other to establish an SSL link, when
two-way SSL is enabled. Both sides now know the identity of the other from their respective certificates.",
"propertyGroup": "CONNECTION_PROPS",
"propertyName": "enableTwoWaySSL",
"propertyShortDesc": "Select \"Yes\" if both the client and server should authenticate each other.",
"propertyType": "CHOICE",
"requiredFlag": false
},
{
    "displayName": "Identity keystore alias name (Optional)",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Both the client and the server pass certificates to each other to establish an SSL link, when
two-way SSL is enabled. Name of the identity certificate alias to use for that communication. This value should match the alias
that was provided to import identity to ICS.",
    "propertyGroup": "CONNECTION_PROPS",
    "propertyName": "sslCertificateAlias",
    "propertyShortDesc": "Name of alias to use for establishing identity during two way SSL communication",
    "propertyType": "STRING",
    "requiredFlag": false
}
],
"created": "2023-11-13T15:58:08.535+0000",
"createdBy": "peter.obert@oracle.com",
"description": "Invoking the process automation to create new process instances",
"hasOverride": false,
"id": "POB_OICTOOPA_INVOKE",
"keywords": "",
"lastRefreshedDate": "2025-02-21T06:51:49.587+0000",
"lastUpdated": "2025-03-20T15:00:30.388+0000",
"lastUpdatedBy": "peter.obert@oracle.com",

```



```

    "links": [
      {
        "href": "https://design.integration.<<your-OIC-Instance-
region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/POB_OICTOOPA_INVOKE?integrationInstance=teamoic3-<<Storage-
namespace>>-fr",
        "rel": "self"
      },
      {
        "href": "https://design.integration.<<your-OIC-Instance-
region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/POB_OICTOOPA_INVOKE?integrationInstance=teamoic3-<<Storage-
namespace>>-fr",
        "rel": "canonical"
      }
    ],
    "lockedFlag": false,
    "metadataDownloadState": "COMPLETE",
    "metadataDownloadSupportedFlag": true,
    "name": "pob-OICtoOPA-Invoke",
    "percentageComplete": 100,
    "privateEndpoint": false,
    "privateEndpointSupported": true,
    "refreshedStatus": "COMPLETE",
    "role": "TARGET",
    "securityPolicy": "OAUTH_RESOURCE_OWNER_PASSWORD_CREDENTIALS",
    "securityPolicyInfo": [
      {
        "scope": "TARGET",
        "securityPolicy": "OCI_SERVICE_INVOCATION",
        "securityProperties": [
          {
            "defaultValue": "DefaultValueForRPST",
            "hasAttachment": false,
            "hiddenFlag": true,
            "propertyGroup": "CREDENTIALS",
            "propertyName": "dummy_credential_rpst",
            "propertyType": "STRING",
            "requiredFlag": true
          }
        ]
      },
      {
        "scope": "TARGET",
        "securityPolicy": "OCI_SIGNATURE_VERSION1",
        "securityProperties": [

```

```
{
  "displayName": "Tenancy OCID",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyGroup": "CREDENTIALS",
  "propertyName": "TenancyOCID",
  "propertyShortDesc": "Tenancy OCID of OCI",
  "propertyType": "STRING",
  "requiredFlag": true
},
{
  "displayName": "User OCID",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "User OCID of OCI",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "UserOCID",
  "propertyShortDesc": "User OCID of OCI",
  "propertyType": "STRING",
  "requiredFlag": true
},
{
  "displayName": "Private Key",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "OCI Private Key",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "PrivateKey",
  "propertyShortDesc": "OCI Private Key",
  "propertyType": "URL_OR_FILE",
  "requiredFlag": true
},
{
  "displayName": "Finger Print",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "Finger Print",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "FingerPrint",
  "propertyShortDesc": "Finger Print",
  "propertyType": "STRING",
  "requiredFlag": true
},
{
```

```

        "displayName": "Pass Phrase",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Password entered at the time of generating the public/private key.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "PassPhrase",
        "propertyShortDesc": "Password entered at the time of generating the public/private key.",
        "propertyType": "PASSWORD",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "JWT_USER_ASSERTION_FOR_OAUTH",
    "securityProperties": [
        {
            "displayName": "Access Token URI",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.access.token.uri",
            "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "JWT headers in json format",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "JWT headers in json format",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "jwt_header",
            "propertyShortDesc": "JWT headers in json format",
            "propertyType": "URL_OR_FILE",
            "requiredFlag": true
        },
        {
            "displayName": "JWT payload in json format",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "JWT payload in json format",
            "propertyGroup": "CREDENTIALS",

```

```

        "propertyName": "jwt_payload",
        "propertyShortDesc": "JWT payload in json format",
        "propertyType": "URL_OR_FILE",
        "requiredFlag": true
    },
    {
        "displayName": "JWT Private Key Alias",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "JWT Private Key Alias",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "jwt_signing_key_alias_name",
        "propertyShortDesc": "JWT Private Key Alias",
        "propertyType": "STRING",
        "requiredFlag": true
    },
    {
        "displayName": "Scope",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Scope",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "scope",
        "propertyShortDesc": "Scope",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Access Token Request",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Default: -X POST -H 'Content-Type: application/x-www-form-urlencoded' -d
'grant_type=urn:ietf:params:oauth:grant-type:jwt-
bearer&assertion=${user_assertion}&client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-
bearer&client_assertion=${client_assertion}&scope=${scope}' ${access_token_uri}?<query params> . Here scope is optional",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "jwtAccessTokenRequest",
        "propertyShortDesc": "Access Token Request that should be used to fetch the access token.",
        "propertyType": "STRING",
        "requiredFlag": false
    }
]
},
{

```

```

"scope": "TARGET",
"securityPolicy": "CUSTOM_SINGLE_TOKEN",
"securityProperties": [
  {
    "displayName": "API Key",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Generated API Key used to identify the client that is making the request.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "accessToken",
    "propertyShortDesc": "Generated API key.",
    "propertyType": "PASSWORD",
    "requiredFlag": true
  },
  {
    "displayName": "API Key Usage",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "API Key Usage. URI syntax to illustrate how the API KEY should be passed to access a
protected resource. Example: To pass the API-KEY as a query parameter \"?<query-param-name>=${api-key}\" or To pass the API-KEY as
a Header \"-H Authorization: Bearer ${api-key}\" without quotes.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "accessTokenUsage",
    "propertyShortDesc": "Default: -H Authorization: Bearer ${api-key}",
    "propertyType": "TEXT",
    "requiredFlag": false
  }
]
},
{
  "scope": "TARGET",
  "securityPolicy": "OAUTH_AUTHORIZATION_CODE_CREDENTIALS",
  "securityProperties": [
    {
      "displayName": "Client Id",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.client.id",
      "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
      "propertyType": "STRING",
      "requiredFlag": true
    }
  ]
}

```

```

    },
    {
      "displayName": "Client Secret",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.client.secret",
      "propertyShortDesc": "<unique random string matches your API console project>",
      "propertyType": "PASSWORD",
      "requiredFlag": true
    },
    {
      "displayName": "Authorization Code URI",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "This endpoint is the target of the initial request. It handles active session lookup,
authenticating the user, and user consent.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.auth.code.uri",
      "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/auth",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Access Token URI",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.access.token.uri",
      "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Scope",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Permissions your application is requesting on behalf of the user.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.scope",
      "propertyShortDesc": "for example: read,write.",
    }
  ]
}

```

```

        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Send client credentials in body",
                "keyName": "client_credentials_in_body",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "Send client credentials as basic auth header",
                "keyName": "client_credentials_as_header",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "displayName": "Client Authentication",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Client Authentication",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "clientAuthentication",
        "propertyShortDesc": "Client Authentication",
        "propertyType": "CHOICE",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "CUSTOM_THREE_LEGGED",
    "securityProperties": [
        {
            "displayName": "Authorization Request",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Authorization url where the user will be redirected to provide consent.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "authRequest",
            "propertyShortDesc": "Example : <authUri>?<queryParams>",
            "propertyType": "TEXT",
            "requiredFlag": true
        }
    ],
    {

```

```

      "displayName": "Access Token Request",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Access Token Request that should be used to fetch the access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "accessTokenRequest",
      "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <access-token-uri><query params>",
      "propertyType": "TEXT",
      "requiredFlag": true
    },
    {
      "displayName": "Refresh Token Request",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Refresh Token Request that should be used to fetch the access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "refreshTokenRequest",
      "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <refresh-token-uri><query params>",
      "propertyType": "TEXT",
      "requiredFlag": false
    },
    {
      "displayName": "$auth_code",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Regex that identifies the authorization code.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "$auth_code",
      "propertyShortDesc": "Default: code",
      "propertyType": "STRING",
      "requiredFlag": false
    },
    {
      "displayName": "$access_token",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Regex that identifies the access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "$access_token",
      "propertyShortDesc": "Default: access.[tT]oken",
      "propertyType": "STRING",
      "requiredFlag": false
    }
  ],
  {

```



```

        "displayName": "$refresh_token",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Regex that identifies the refresh token.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "$refresh_token",
        "propertyShortDesc": "Default: refresh.[tT]oken",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "$expiry",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Numeric value (in seconds)that specifies the expiry interval or a regex that
identifies when the access token expires.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "$expiry",
        "propertyShortDesc": "Default: expires_in",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "$token_type",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Regex that identifies the access token type.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "$token_type",
        "propertyShortDesc": "Default: token.?[tT]ype",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "access_token_usage",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Access token usage. A curl type syntax to illustrate how access token should be passed
to access a protected resource.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "accessTokenUsage",
        "propertyShortDesc": "Default: -H Authorization ${token_type} ${access_token}",
        "propertyType": "TEXT",
        "requiredFlag": false
    }

```

```

    }
  ]
},
{
  "scope": "TARGET",
  "securityPolicy": "OAUTH_ONE_TOKEN_BASED",
  "securityProperties": [
    {
      "displayName": "Consumer Key",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to identify the client that is making the request.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth_consumer_key",
      "propertyShortDesc": "Registered consumer key.",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Consumer Secret",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to authorize the client that is making the request.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth_consumer_secret",
      "propertyShortDesc": "Registered consumer Secret.",
      "propertyType": "PASSWORD",
      "requiredFlag": true
    },
    {
      "displayName": "Token",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to access protected resource.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth_token",
      "propertyShortDesc": "Registered token",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Token Secret",
      "hasAttachment": false,
      "hiddenFlag": false,

```

```

        "propertyDescription": "Used for generating signature for the request.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth_token_secret",
        "propertyShortDesc": "Token secret provided by server.",
        "propertyType": "PASSWORD",
        "requiredFlag": true
    },
    {
        "displayName": "Realm",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Used for identifying the account.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "realm",
        "propertyShortDesc": "Account identifier.",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "acceptableKeys": [
            {
                "displayName": "HMAC-SHA1",
                "keyName": "HMAC-SHA1",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "HMAC-SHA256",
                "keyName": "HMAC-SHA256",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "defaultPropertyValue": "HMAC-SHA1",
        "displayName": "Signature Method (Default Value:HMAC-SHA1)",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Used for signing the request.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth_signature_method",
        "propertyShortDesc": "Signature Method used to sign the request.",
        "propertyType": "CHOICE",
        "requiredFlag": false
    }
]
},

```

```

{
  "scope": "TARGET",
  "securityPolicy": "OAUTH_RESOURCE_OWNER_PASSWORD_CREDENTIALS",
  "securityProperties": [
    {
      "displayName": "Access Token URI",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.access.token.uri",
      "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Client Id",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.client.id",
      "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Client Secret",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.client.secret",
      "propertyShortDesc": "<unique random string matches your API console project>",
      "propertyType": "PASSWORD",
      "requiredFlag": true
    },
    {
      "displayName": "Scope",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Permissions your application is requesting on behalf of the user.",

```

```

    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.scope",
    "propertyShortDesc": "for example: read,write.",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "Auth Request Media Type",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Auth Content-Type",
    "propertyGroup": "OTHER",
    "propertyName": "oauth.request.content.type",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "Username",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "A username credential",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "username",
    "propertyShortDesc": "Use UserName",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Password",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "A password credential",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "password",
    "propertyShortDesc": "Enter Password",
    "propertyType": "PASSWORD",
    "requiredFlag": true
  },
  {
    "acceptableKeys": [
      {
        "displayName": "Send client credentials as basic auth header",
        "keyName": "client_credentials_as_header",
        "scope": "SOURCE_AND_TARGET"
      }
    ]
  }

```

```

    },
    {
        "displayName": "Send client credentials in body",
        "keyName": "client_credentials_in_body",
        "scope": "SOURCE_AND_TARGET"
    }
],
"displayName": "Client Authentication",
"hasAttachment": false,
"hiddenFlag": false,
"propertyDescription": "Client Authentication",
"propertyGroup": "CREDENTIALS",
"propertyName": "clientAuthentication",
"propertyShortDesc": "Client Authentication",
"propertyType": "CHOICE",
"requiredFlag": false
}
]
},
{
    "scope": "TARGET",
    "securityPolicy": "CUSTOM_TWO_LEGGED",
    "securityProperties": [
        {
            "displayName": "Access Token Request",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Access Token Request that should be used to fetch the access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "accessTokenRequest",
            "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <access-token-uri>?<query params>",
            "propertyType": "TEXT",
            "requiredFlag": true
        },
        {
            "displayName": "Refresh Token Request",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Refresh Token Request that should be used to fetch the access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "refreshTokenRequest",
            "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <refresh-token-uri>?<query params>",
            "propertyType": "TEXT",
            "requiredFlag": false
        }
    ]
}

```

```

    },
    {
      "displayName": "$access_token",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Regex that identifies the access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "$access_token",
      "propertyShortDesc": "Default: access.[tT]oken",
      "propertyType": "STRING",
      "requiredFlag": false
    },
    {
      "displayName": "$refresh_token",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Regex that identifies the refresh token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "$refresh_token",
      "propertyShortDesc": "Default: refresh.[tT]oken",
      "propertyType": "STRING",
      "requiredFlag": false
    },
    {
      "displayName": "$expiry",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Numeric value (in seconds)that specifies the expiry interval or a regex that
identifies when the access token expires.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "$expiry",
      "propertyShortDesc": "Default: expires_in",
      "propertyType": "STRING",
      "requiredFlag": false
    },
    {
      "displayName": "$token_type",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Regex that identifies the access token type.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "$token_type",
      "propertyShortDesc": "Default: token.?[tT]ype",
      "propertyType": "STRING",
    }
  ]
}

```

```

        "requiredFlag": false
    },
    {
        "displayName": "access_token_usage",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Access token usage. A curl type syntax to illustrate how access token should be passed
to access a protected resource.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "accessTokenUsage",
        "propertyShortDesc": "Default: -H Authorization ${token_type} ${access_token}",
        "propertyType": "TEXT",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "GENERIC_OAUTH_AUTHORIZATION_CODE",
    "securityProperties": [
        {
            "displayName": "Client Id",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.client.id",
            "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "Client Secret",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.client.secret",
            "propertyShortDesc": "<unique random string matches your API console project>",
            "propertyType": "PASSWORD",
            "requiredFlag": true
        }
    ]
},

```



```

{
  "displayName": "Authorization Code URI",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "This endpoint is the target of the initial request. It handles active session lookup,
authenticating the user, and user consent.",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "oauth.auth.code.uri",
  "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/auth",
  "propertyType": "STRING",
  "requiredFlag": true
},
{
  "displayName": "Access Token URI",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "oauth.access.token.uri",
  "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
  "propertyType": "STRING",
  "requiredFlag": true
},
{
  "displayName": "Scope",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "Permissions your application is requesting on behalf of the user.",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "oauth.scope",
  "propertyShortDesc": "for example: read,write.",
  "propertyType": "STRING",
  "requiredFlag": false
},
{
  "acceptableKeys": [
    {
      "displayName": "Send client credentials in body",
      "keyName": "client_credentials_in_body",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Send client credentials as basic auth header",
      "keyName": "client_credentials_as_header",

```

```

        "scope": "SOURCE_AND_TARGET"
    }
],
    "displayName": "Client Authentication",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Client Authentication",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "clientAuthentication",
    "propertyShortDesc": "Client Authentication",
    "propertyType": "CHOICE",
    "requiredFlag": false
}
]
},
{
    "scope": "TARGET",
    "securityPolicy": "JWT_CLIENT_CREDENTIALS_ASSERTION_FOR_OAUTH",
    "securityProperties": [
        {
            "displayName": "Access Token URI",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.access.token.uri",
            "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "JWT headers in json format",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "JWT headers in json format",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "jwt_header",
            "propertyShortDesc": "JWT headers in json format",
            "propertyType": "URL_OR_FILE",
            "requiredFlag": true
        },
        {
            "displayName": "JWT payload in json format",
            "hasAttachment": false,

```

```

        "hiddenFlag": false,
        "propertyDescription": "JWT payload in json format",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "jwt_payload",
        "propertyShortDesc": "JWT payload in json format",
        "propertyType": "URL_OR_FILE",
        "requiredFlag": true
    },
    {
        "displayName": "JWT Private Key Alias",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "JWT Private Key Alias",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "jwt_signing_key_alias_name",
        "propertyShortDesc": "JWT Private Key Alias",
        "propertyType": "STRING",
        "requiredFlag": true
    },
    {
        "displayName": "Scope",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Scope",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "scope",
        "propertyShortDesc": "Scope",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Access Token Request",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Default: -X POST -H 'Content-Type: application/x-www-form-urlencoded' -d
'grant_type=client_credentials&client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-
bearer&client_assertion=${client_assertion}&scope=${scope}' ${access_token_uri}?<query params> . Here scope is optional",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "jwtAccessTokenRequest",
        "propertyShortDesc": "Access Token Request that should be used to fetch the access token.",
        "propertyType": "STRING",
        "requiredFlag": false
    }
]

```

```

    },
    {
      "scope": "TARGET",
      "securityPolicy": "OAUTH_CLIENT_CREDENTIALS",
      "securityProperties": [
        {
          "displayName": "Access Token URI",
          "hasAttachment": false,
          "hiddenFlag": false,
          "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
          "propertyGroup": "CREDENTIALS",
          "propertyName": "oauth.access.token.uri",
          "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
          "propertyType": "STRING",
          "requiredFlag": true
        },
        {
          "displayName": "Client Id",
          "hasAttachment": false,
          "hiddenFlag": false,
          "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
          "propertyGroup": "CREDENTIALS",
          "propertyName": "oauth.client.id",
          "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
          "propertyType": "STRING",
          "requiredFlag": true
        },
        {
          "displayName": "Client Secret",
          "hasAttachment": false,
          "hiddenFlag": false,
          "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
          "propertyGroup": "CREDENTIALS",
          "propertyName": "oauth.client.secret",
          "propertyShortDesc": "<unique random string matches your API console project>",
          "propertyType": "PASSWORD",
          "requiredFlag": true
        },
        {
          "displayName": "Scope",
          "hasAttachment": false,
          "hiddenFlag": false,

```

```

        "propertyDescription": "Permissions your application is requesting on behalf of the user.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.scope",
        "propertyShortDesc": "for example: read,write.",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Auth Request Media Type",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "application/json or application/x-www-form-urlencoded",
        "propertyGroup": "OTHER",
        "propertyName": "oauth.request.content.type",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Send client credentials as basic auth header",
                "keyName": "client_credentials_as_header",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "Send client credentials in body",
                "keyName": "client_credentials_in_body",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "displayName": "Client Authentication",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Client Authentication",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "clientAuthentication",
        "propertyShortDesc": "Client Authentication",
        "propertyType": "CHOICE",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",

```

```

"securityPolicy": "AWS_SIGNATURE_VERSION4",
"securityProperties": [
  {
    "displayName": "Access Key",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyGroup": "CREDENTIALS",
    "propertyName": "AccessKey",
    "propertyShortDesc": "Access Key of AWS",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Secret Key",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Secret Key of AWS",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "SecretKey",
    "propertyShortDesc": "Secret Key of AWS",
    "propertyType": "PASSWORD",
    "requiredFlag": true
  },
  {
    "acceptableKeys": [
      {
        "displayName": "US West (Oregon) Region (us-west-2)",
        "keyName": "us-west-2",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "US West (N. California) Region (us-west-1)",
        "keyName": "us-west-1",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "US East (Ohio) Region (us-east-2)",
        "keyName": "us-east-2",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "US East (N. Virginia) Region (us-east-1)",
        "keyName": "us-east-1",
        "scope": "SOURCE_AND_TARGET"
      }
    ]
  }
]

```

```

    },
    {
      "displayName": "Asia Pacific (Mumbai) Region (ap-south-1)",
      "keyName": "ap-south-1",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Asia Pacific (Seoul) Region (ap-northeast-2)",
      "keyName": "ap-northeast-2",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Asia Pacific (Tokyo) Region (ap-northeast-1)",
      "keyName": "ap-northeast-1",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Asia Pacific (Sydney) Region (ap-southeast-2)",
      "keyName": "ap-southeast-2",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Asia Pacific (Singapore) Region (ap-southeast-1)",
      "keyName": "ap-southeast-1",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Canada (Central) Region (ca-central-1)",
      "keyName": "ca-central-1",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "China (Beijing) Region (cn-north-1)",
      "keyName": "cn-north-1",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "EU (Frankfurt) Region (eu-central-1)",
      "keyName": "eu-central-1",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "EU (Ireland) Region (eu-west-1)",
      "keyName": "eu-west-1",

```

```

    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "EU (London) Region (eu-west-2)",
    "keyName": "eu-west-2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "EU (Paris) Region (eu-west-3)",
    "keyName": "eu-west-3",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "South America (São Paulo) Region (sa-east-1)",
    "keyName": "sa-east-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS GovCloud (US) (us-gov-west-1)",
    "keyName": "us-gov-west-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Africa (Cape Town) Region (af-south-1)",
    "keyName": "af-south-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Hong Kong) Region (ap-east-1)",
    "keyName": "ap-east-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Hyderabad) Region (ap-south-2)",
    "keyName": "ap-south-2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Jakarta) Region (ap-southeast-3)",
    "keyName": "ap-southeast-3",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Melbourne) Region (ap-southeast-4)",

```



```

        "keyName": "ap-southeast-4",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Asia Pacific (Osaka) Region (ap-northeast-3)",
        "keyName": "ap-northeast-3",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Europe (Milan) Region (eu-south-1)",
        "keyName": "eu-south-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Europe (Spain) Region (eu-south-2)",
        "keyName": "eu-south-2",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Europe (Stockholm) Region (eu-north-1)",
        "keyName": "eu-north-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Europe (Zurich) Region (eu-central-2)",
        "keyName": "eu-central-2",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Israel (Tel Aviv) Region (il-central-1)",
        "keyName": "il-central-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Middle East (Bahrain) Region (me-south-1)",
        "keyName": "me-south-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Middle East (UAE) (me-central-1)",
        "keyName": "me-central-1",
        "scope": "SOURCE_AND_TARGET"
    }
],

```

```

    "displayName": "AWS Region",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "AWS Region",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "AWSRegion",
    "propertyShortDesc": "AWS Region",
    "propertyType": "CHOICE",
    "requiredFlag": true
  },
  {
    "acceptableKeys": [
      {
        "displayName": "Alexa for Business",
        "keyName": "a4b",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "Amazon API Gateway",
        "keyName": "apigateway",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "Amazon AppStream 2.0",
        "keyName": "appstream2",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "AWS AppSync Control Plane",
        "keyName": "appsync",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "AWS AppSync Data Plane",
        "keyName": "appsync-api",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "Athena",
        "keyName": "athena",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "AWS Auto Scaling",

```

```

        "keyName": "autoscaling",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Batch",
        "keyName": "batch",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Cost Explorer",
        "keyName": "ce",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Cost and Usage Reports",
        "keyName": "cur",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Budgets",
        "keyName": "budgets",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Price List Service",
        "keyName": "api.pricing",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Certificate Manager",
        "keyName": "acm",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Certificate Manager Private Certificate Authority",
        "keyName": "acm-pca",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Cloud9",
        "keyName": "cloud9",
        "scope": "SOURCE_AND_TARGET"
    },
    {

```

```
    "displayName": "Amazon Cloud Directory",
    "keyName": "clouddirectory",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CloudFormation",
    "keyName": "cloudformation",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon CloudFront",
    "keyName": "cloudfront",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CloudHSM",
    "keyName": "cloudhsmv2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CloudHSM Classic",
    "keyName": "cloudhsm",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon CloudSearch",
    "keyName": "cloudsearch",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CloudTrail",
    "keyName": "cloudtrail",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon CloudWatch",
    "keyName": "monitoring",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon CloudWatch Events",
    "keyName": "events",
    "scope": "SOURCE_AND_TARGET"
  },
  },
```

```
{
  "displayName": "Amazon Execute API",
  "keyName": "execute-api",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon CloudWatch Logs",
  "keyName": "logs",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS CodeBuild",
  "keyName": "codebuild",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS CodeCommit",
  "keyName": "codecommit",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS CodeDeploy",
  "keyName": "codedeploy",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS CodePipeline",
  "keyName": "codepipeline",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS CodeStar",
  "keyName": "codestar",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Cognito Your User Pools",
  "keyName": "cognito-idp",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Cognito Federated Identities",
  "keyName": "cognito-identity",
  "scope": "SOURCE_AND_TARGET"
}
```

```
    },
    {
      "displayName": "Amazon Cognito Sync",
      "keyName": "cognito-sync",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Comprehend",
      "keyName": "Comprehend",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Config",
      "keyName": "config",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Config Rules",
      "keyName": "configrules",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Connect",
      "keyName": "connect",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Data Pipeline",
      "keyName": "datapipeline",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Database Migration Service",
      "keyName": "dms",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS DeepLens",
      "keyName": "deeplens",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Device Farm",
      "keyName": "devicefarm",
```

```

    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Direct Connect",
    "keyName": "directconnect",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Directory Service",
    "keyName": "ds",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon DynamoDB",
    "keyName": "dynamodb",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "DynamoDB Accelerator (DAX)",
    "keyName": "dax",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon DynamoDB Streams",
    "keyName": "streams.dynamodb",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elastic Beanstalk",
    "keyName": "elasticbeanstalk",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elastic Beanstalk Health Service",
    "keyName": "elasticbeanstalk-health",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic Compute Cloud (Amazon EC2)",
    "keyName": "ec2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic Container Service",

```

```
    "keyName": "ecr",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic Container Service",
    "keyName": "ecs",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic Container Service for Kubernetes (Amazon EKS)",
    "keyName": "eks",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic File System",
    "keyName": "elasticfilesystem",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Elastic Load Balancing",
    "keyName": "elasticloadbalancing",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic Transcoder",
    "keyName": "elastictranscoder",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon ElastiCache",
    "keyName": "elasticache",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elasticsearch Service",
    "keyName": "es",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon EMR",
    "keyName": "elasticmapreduce",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    
```



```
    "displayName": "AWS Firewall Manager",
    "keyName": "fms",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon GameLift",
    "keyName": "gamelift",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Glacier",
    "keyName": "glacier",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Glue",
    "keyName": "glue",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Greengrass",
    "keyName": "greengrass",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon GuardDuty",
    "keyName": "guardduty",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Health",
    "keyName": "health",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Identity and Access Management (IAM)",
    "keyName": "iam",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Import/Export",
    "keyName": "importexport",
    "scope": "SOURCE_AND_TARGET"
  },
  },
```

```
{
  "displayName": "Amazon Inspector",
  "keyName": "inspector",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS IoT 1-Click",
  "keyName": "iot1click",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS IoT Analytics",
  "keyName": "iotanalytics",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS IoT Core",
  "keyName": "iot",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Key Management Service",
  "keyName": "kms",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Kinesis Data Analytics",
  "keyName": "kinesisanalytics",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Kinesis Data Firehose",
  "keyName": "firehose",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Kinesis Data Streams",
  "keyName": "kinesis",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Kinesis Video Streams",
  "keyName": "kinesisvideo",
  "scope": "SOURCE_AND_TARGET"
}
```

```
    },
    {
      "displayName": "AWS Lambda",
      "keyName": "lambda",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Lex (Model Building Endpoints)",
      "keyName": "models.lex",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Lex (Runtime Endpoints)",
      "keyName": "runtime.lex",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Lightsail",
      "keyName": "lightsail",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Macie",
      "keyName": "macie",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Machine Learning",
      "keyName": "machinelearning",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Elemental MediaConvert",
      "keyName": "mediaconvert",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Elemental MediaLive",
      "keyName": "medialive",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Elemental MediaPackage",
      "keyName": "mediapackage",
```

```

    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaStore",
    "keyName": "mediastore",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaTailor",
    "keyName": "mediatailor",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Migration Hub",
    "keyName": "mgh",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Mobile Analytics",
    "keyName": "mobileanalytics",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon MQ",
    "keyName": "mq",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS OpsWorks (CM)",
    "keyName": "opsworks-cm",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS OpsWorks (Stacks)",
    "keyName": "opsworks",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Organizations",
    "keyName": "organizations",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Pinpoint",

```

```
    "keyName": "pinpoint",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Polly",
    "keyName": "polly",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon QuickSight",
    "keyName": "quicksight",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Redshift",
    "keyName": "redshift",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Rekognition",
    "keyName": "rekognition",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Relational Database Service (Amazon RDS)",
    "keyName": "rds",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Resource Groups",
    "keyName": "resource-groups",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Secrets Manager",
    "keyName": "secretsmanager",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Server Migration Service",
    "keyName": "sms",
    "scope": "SOURCE_AND_TARGET"
  },
  {

```

```
    "displayName": "Amazon SageMaker",
    "keyName": "api.sagemaker",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Security Token Service (AWS STS)",
    "keyName": "sts",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Serverless Application Repository",
    "keyName": "serverlessrepo",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Service Catalog",
    "keyName": "servicecatalog",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Shield Advanced",
    "keyName": "Shield",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Simple Email Service (Amazon SES)",
    "keyName": "email-smtp",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Simple Notification Service (Amazon SNS)",
    "keyName": "sns",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Simple Queue Service (Amazon SQS)",
    "keyName": "sqs",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Simple Storage Service (Amazon S3)",
    "keyName": "s3",
    "scope": "SOURCE_AND_TARGET"
  },
}
```

```
{
  "displayName": "Amazon Simple Storage Service Website Endpoints",
  "keyName": "s3-website",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Simple Workflow Service (Amazon SWF)",
  "keyName": "swf",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon SimpleDB",
  "keyName": "sdb",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Snowball",
  "keyName": "snowball",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Step Functions",
  "keyName": "states",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Storage Gateway",
  "keyName": "storagegateway",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Systems Manager",
  "keyName": "ssm",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Transcribe",
  "keyName": "transcribe",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Translate",
  "keyName": "Translate",
  "scope": "SOURCE_AND_TARGET"
}
```

```

    },
    {
      "displayName": "Amazon Translate (FIPS)",
      "keyName": "translate-fips",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon VPC",
      "keyName": "waf",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon WorkDocs",
      "keyName": "workdocs",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon WorkSpaces",
      "keyName": "workspaces",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS X-Ray",
      "keyName": "xray",
      "scope": "SOURCE_AND_TARGET"
    }
  ],
  "displayName": "Service Name",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "Service Name like S3",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "ServiceName",
  "propertyShortDesc": "Service Name",
  "propertyType": "CHOICE",
  "requiredFlag": true
}
]
},
{
  "scope": "TARGET",
  "securityPolicy": "NONE",
  "securityProperties": []
},

```



```

{
  "scope": "SOURCE_AND_TARGET",
  "securityPolicy": "BASIC_AUTH",
  "securityProperties": [
    {
      "displayName": "Username",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "A username credential",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "username",
      "propertyShortDesc": "Use UserName",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Password",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "A password credential",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "password",
      "propertyShortDesc": "Enter Password",
      "propertyType": "PASSWORD",
      "requiredFlag": true
    }
  ]
},
{
  "securityProperties": [
    {
      "displayName": "Access Token URI",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.access.token.uri",
      "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
      "propertyType": "STRING",
      "propertyValue": "https://idcs-5ba32fa3496f48289532f8fc10f47032.identity.oraclecloud.com:443/oauth2/v1/token",
      "requiredFlag": true
    },
    {
      "displayName": "Client Id",

```

```

        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Used to identify the client(the software requesting an access token) that is making the
request. The value passed in this parameter must exactly match the value shown in your API console project.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.client.id",
        "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
        "propertyType": "STRING",
        "propertyValue": "<<your-IntegratedApp-ClientID>>",
        "requiredFlag": true
    },
    {
        "displayName": "Client Secret",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making the
request. The value passed in this parameter must exactly match the value shown in your API console project.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.client.secret",
        "propertyShortDesc": "<unique random string matches your API console project>",
        "propertyType": "PASSWORD",
        "propertyValue": "*****",
        "requiredFlag": true
    },
    {
        "displayName": "Scope",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Permissions your application is requesting on behalf of the user.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.scope",
        "propertyShortDesc": "for example: read,write.",
        "propertyType": "STRING",
        "propertyValue": "https://opa-oic-teamoic3-1740047240850-<<Storage-namespace>>-
fra.process.oci.oraclecloud.com/process",
        "requiredFlag": false
    },
    {
        "displayName": "Auth Request Media Type",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Auth Content-Type",
        "propertyGroup": "OTHER",
        "propertyName": "oauth.request.content.type",

```

```

    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "Username",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "A username credential",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "username",
    "propertyShortDesc": "Use UserName",
    "propertyType": "STRING",
    "propertyValue": "peter.obert+local@oracle.com",
    "requiredFlag": true
  },
  {
    "displayName": "Password",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "A password credential",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "password",
    "propertyShortDesc": "Enter Password",
    "propertyType": "PASSWORD",
    "propertyValue": "*****",
    "requiredFlag": true
  },
  {
    "acceptableKeys": [
      {
        "displayName": "Send client credentials as basic auth header",
        "keyName": "client_credentials_as_header",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "Send client credentials in body",
        "keyName": "client_credentials_in_body",
        "scope": "SOURCE_AND_TARGET"
      }
    ],
    "displayName": "Client Authentication",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Client Authentication",

```

```
        "propertyGroup": "CREDENTIALS",
        "propertyName": "clientAuthentication",
        "propertyShortDesc": "Client Authentication",
        "propertyType": "CHOICE",
        "propertyValue": "client_credentials_as_header",
        "requiredFlag": false
    },
    ],
    "status": "CONFIGURED",
    "testStatus": false,
    "usage": 4,
    "usageActive": 1
}
```

POSTOIC3_DEV_UPDATEconnection

[Open request](#)

https://design.integration.<<your-OIC-Instance-region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/?integrationInstance=<<your-OIC-InstanceName>>

Add request description...

Authorization

OAuth 2.0

Query Params

integrationInstance

<<your-OIC-InstanceName>>

Bodyraw (json)

View More

json

```
{
  "adapterType": {
    "displayName": "REST",
    "links": [],
    "name": "rest",
    "type": "PREINSTALLED"
  },
  "agentRequired": false,
  "agentSupported": true,
```

```
"connectionProperties": [  
  {  
    "acceptableKeys": [  
      {  
        "displayName": "REST API Base URL",  
        "keyName": "restUrl",  
        "scope": "SOURCE_AND_TARGET"  
      },  
      {  
        "displayName": "Open API (1.0/2.0/3.0) URL",  
        "keyName": "openApiUrl",  
        "scope": "SOURCE_AND_TARGET"  
      }  
    ],  
    "displayName": "Connection Type",  
    "hasAttachment": false,  
    "hiddenFlag": false,  
    "propertyGroup": "CONNECTION_PROPS",  
    "propertyName": "connectionType",  
    "propertyType": "CHOICE",  
    "propertyValue": "restUrl",  
    "requiredFlag": true  
  },  
  {  
    "acceptableKeys": [  
      {  
        "displayName": "TLSv1.1",  
        "keyName": "TLSv1.1",  
        "scope": "SOURCE_AND_TARGET"  
      },  
      {  
        "displayName": "TLSv1.2",  
        "keyName": "TLSv1.2",  
        "scope": "SOURCE_AND_TARGET"  
      }  
    ],  
    "displayName": "TLS Version",  
    "hasAttachment": false,  
    "hiddenFlag": false,  
    "propertyGroup": "CONNECTION_PROPS",  
    "propertyName": "tlsVersion",  
    "propertyShortDesc": "TLS Version",  
    "propertyType": "CHOICE",  
    "requiredFlag": false  
  }  
]
```

```

    },
    {
      "displayName": "Connection URL",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyGroup": "CONNECTION_PROPS",
      "propertyName": "connectionUrl",
      "propertyShortDesc": "Please make sure that this value really corresponds to the type selected above.",
      "propertyType": "URL",
      "propertyValue": "https://opa-oic-teamoic3-1740047240850-<<Storage-namespace>>-fra.process.oci.oraclecloud.com/process/api",
      "requiredFlag": true
    },
    {
      "acceptableKeys": [
        {
          "displayName": "Yes",
          "keyName": "true",
          "scope": "SOURCE_AND_TARGET"
        },
        {
          "displayName": "No",
          "keyName": "false",
          "scope": "SOURCE_AND_TARGET"
        }
      ],
      "displayName": "Enable two way SSL for outbound connections (Optional)",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Both the client and the server pass certificates to each other to establish an SSL link, when two-way SSL is enabled. Both sides now know the identity of the other from their respective certificates.",
      "propertyGroup": "CONNECTION_PROPS",
      "propertyName": "enableTwoWaySSL",
      "propertyShortDesc": "Select \"Yes\" if both the client and server should authenticate each other.",
      "propertyType": "CHOICE",
      "requiredFlag": false
    },
    {
      "displayName": "Identity keystore alias name (Optional)",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Both the client and the server pass certificates to each other to establish an SSL link, when two-way SSL is enabled. Name of the identity certificate alias to use for that communication. This value should match the alias that was provided to import identity to ICS.",
    }
  ]
}

```

```

        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "sslCertificateAlias",
        "propertyShortDesc": "Name of alias to use for establishing identity during two way SSL communication",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    ],
    "created": "2023-11-13T15:58:08.535+0000",
    "createdBy": "peter.obert@oracle.com",
    "description": "Invoking the process automation to create new process instances",
    "hasOverride": false,
    "id": "POB_OICTOOPA_INVOKE",
    "keywords": "",
    "lastRefreshedDate": "2025-02-21T06:51:49.587+0000",
    "lastUpdated": "2025-03-07T13:40:44.908+0000",
    "lastUpdatedBy": "peter.obert@oracle.com",
    "links": [
        {
            "href": "https://design.integration.<<your-OIC-Instance-
region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/POB_OICTOOPA_INVOKE?integrationInstance=teamoic3-<<Storage-
namespace>>-fr",
            "rel": "self"
        },
        {
            "href": "https://design.integration.<<your-OIC-Instance-
region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/POB_OICTOOPA_INVOKE?integrationInstance=teamoic3-<<Storage-
namespace>>-fr",
            "rel": "canonical"
        }
    ],
    "lockedFlag": false,
    "metadataDownloadState": "COMPLETE",
    "metadataDownloadSupportedFlag": true,
    "name": "pob-OICtoOPA-Invoke",
    "percentageComplete": 100,
    "privateEndpoint": false,
    "privateEndpointSupported": true,
    "refreshedStatus": "COMPLETE",
    "role": "TARGET",
    "securityPolicy": "OAUTH_RESOURCE_OWNER_PASSWORD_CREDENTIALS",
    "securityPolicyInfo": [
        {
            "scope": "TARGET",
            "securityPolicy": "OCI_SERVICE_INVOCATION",

```

```

    "securityProperties": [
      {
        "defaultPropertyValue": "DefaultValueForRPST",
        "hasAttachment": false,
        "hiddenFlag": true,
        "propertyGroup": "CREDENTIALS",
        "propertyName": "dummy_credential_rpst",
        "propertyType": "STRING",
        "requiredFlag": true
      }
    ]
  },
  {
    "scope": "TARGET",
    "securityPolicy": "OCI_SIGNATURE_VERSION1",
    "securityProperties": [
      {
        "displayName": "Tenancy OCID",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyGroup": "CREDENTIALS",
        "propertyName": "TenancyOCID",
        "propertyShortDesc": "Tenancy OCID of OCI",
        "propertyType": "STRING",
        "requiredFlag": true
      },
      {
        "displayName": "User OCID",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "User OCID of OCI",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "UserOCID",
        "propertyShortDesc": "User OCID of OCI",
        "propertyType": "STRING",
        "requiredFlag": true
      },
      {
        "displayName": "Private Key",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "OCI Private Key",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "PrivateKey",

```



```

        "propertyShortDesc": "OCI Private Key",
        "propertyType": "URL_OR_FILE",
        "requiredFlag": true
    },
    {
        "displayName": "Finger Print",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Finger Print",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "FingerPrint",
        "propertyShortDesc": "Finger Print",
        "propertyType": "STRING",
        "requiredFlag": true
    },
    {
        "displayName": "Pass Phrase",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Password entered at the time of generating the public/private key.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "PassPhrase",
        "propertyShortDesc": "Password entered at the time of generating the public/private key.",
        "propertyType": "PASSWORD",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "JWT_USER_ASSERTION_FOR_OAUTH",
    "securityProperties": [
        {
            "displayName": "Access Token URI",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.access.token.uri",
            "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
            "propertyType": "STRING",
            "requiredFlag": true
        }
    ]
}

```

```

    "displayName": "JWT headers in json format",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT headers in json format",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_header",
    "propertyShortDesc": "JWT headers in json format",
    "propertyType": "URL_OR_FILE",
    "requiredFlag": true
  },
  {
    "displayName": "JWT payload in json format",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT payload in json format",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_payload",
    "propertyShortDesc": "JWT payload in json format",
    "propertyType": "URL_OR_FILE",
    "requiredFlag": true
  },
  {
    "displayName": "JWT Private Key Alias",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT Private Key Alias",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_signing_key_alias_name",
    "propertyShortDesc": "JWT Private Key Alias",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Scope",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Scope",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "scope",
    "propertyShortDesc": "Scope",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {

```

```

        "displayName": "Access Token Request",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Default: -X POST -H 'Content-Type: application/x-www-form-urlencoded' -d
'grant_type=urn:ietf:params:oauth:grant-type:jwt-
bearer&assertion=${user_assertion}&client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-
bearer&client_assertion=${client_assertion}&scope=${scope}' ${access_token_uri}?<query params> . Here scope is optional",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "jwtAccessTokenRequest",
        "propertyShortDesc": "Access Token Request that should be used to fetch the access token.",
        "propertyType": "STRING",
        "requiredFlag": false
    }
},
{
    "scope": "TARGET",
    "securityPolicy": "CUSTOM_SINGLE_TOKEN",
    "securityProperties": [
        {
            "displayName": "API Key",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Generated API Key used to identify the client that is making the request.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "accessToken",
            "propertyShortDesc": "Generated API key.",
            "propertyType": "PASSWORD",
            "requiredFlag": true
        },
        {
            "displayName": "API Key Usage",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "API Key Usage. URI syntax to illustrate how the API KEY should be passed to access a
protected resource. Example: To pass the API-KEY as a query parameter \"?<query-param-name>=${api-key}\" or To pass the API-KEY as
a Header \"-H Authorization: Bearer ${api-key}\" without quotes.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "accessTokenUsage",
            "propertyShortDesc": "Default: -H Authorization: Bearer ${api-key}",
            "propertyType": "TEXT",
            "requiredFlag": false
        }
    ]
}
]

```

```

    },
    {
      "scope": "TARGET",
      "securityPolicy": "OAUTH_AUTHORIZATION_CODE_CREDENTIALS",
      "securityProperties": [
        {
          "displayName": "Client Id",
          "hasAttachment": false,
          "hiddenFlag": false,
          "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
          "propertyGroup": "CREDENTIALS",
          "propertyName": "oauth.client.id",
          "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
          "propertyType": "STRING",
          "requiredFlag": true
        },
        {
          "displayName": "Client Secret",
          "hasAttachment": false,
          "hiddenFlag": false,
          "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
          "propertyGroup": "CREDENTIALS",
          "propertyName": "oauth.client.secret",
          "propertyShortDesc": "<unique random string matches your API console project>",
          "propertyType": "PASSWORD",
          "requiredFlag": true
        },
        {
          "displayName": "Authorization Code URI",
          "hasAttachment": false,
          "hiddenFlag": false,
          "propertyDescription": "This endpoint is the target of the initial request. It handles active session lookup,
authenticating the user, and user consent.",
          "propertyGroup": "CREDENTIALS",
          "propertyName": "oauth.auth.code.uri",
          "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/auth",
          "propertyType": "STRING",
          "requiredFlag": true
        },
        {
          "displayName": "Access Token URI",
          "hasAttachment": false,

```

```

        "hiddenFlag": false,
        "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.access.token.uri",
        "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
        "propertyType": "STRING",
        "requiredFlag": true
    },
    {
        "displayName": "Scope",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Permissions your application is requesting on behalf of the user.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.scope",
        "propertyShortDesc": "for example: read,write.",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Send client credentials in body",
                "keyName": "client_credentials_in_body",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "Send client credentials as basic auth header",
                "keyName": "client_credentials_as_header",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "displayName": "Client Authentication",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Client Authentication",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "clientAuthentication",
        "propertyShortDesc": "Client Authentication",
        "propertyType": "CHOICE",
        "requiredFlag": false
    }
]
},

```

```

{
  "scope": "TARGET",
  "securityPolicy": "CUSTOM_THREE_LEGGED",
  "securityProperties": [
    {
      "displayName": "Authorization Request",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Authorization url where the user will be redirected to provide consent.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "authRequest",
      "propertyShortDesc": "Example : <authUri>?<queryParams>",
      "propertyType": "TEXT",
      "requiredFlag": true
    },
    {
      "displayName": "Access Token Request",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Access Token Request that should be used to fetch the access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "accessTokenRequest",
      "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <access-token-uri><query params>",
      "propertyType": "TEXT",
      "requiredFlag": true
    },
    {
      "displayName": "Refresh Token Request",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Refresh Token Request that should be used to fetch the access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "refreshTokenRequest",
      "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <refresh-token-uri><query params>",
      "propertyType": "TEXT",
      "requiredFlag": false
    },
    {
      "displayName": "$auth_code",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Regex that identifies the authorization code.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "$auth_code",
    }
  ]
}

```

```

    "propertyShortDesc": "Default: code",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$access_token",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Regex that identifies the access token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$access_token",
    "propertyShortDesc": "Default: access.[tT]oken",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$refresh_token",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Regex that identifies the refresh token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$refresh_token",
    "propertyShortDesc": "Default: refresh.[tT]oken",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$expiry",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Numeric value (in seconds)that specifies the expiry interval or a regex that
identifies when the access token expires.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$expiry",
    "propertyShortDesc": "Default: expires_in",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$token_type",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Regex that identifies the access token type.",
    "propertyGroup": "CREDENTIALS",

```

```

        "propertyName": "$token_type",
        "propertyShortDesc": "Default: token.?[tT]ype",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "access_token_usage",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Access token usage. A curl type syntax to illustrate how access token should be passed
to access a protected resource.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "accessTokenUsage",
        "propertyShortDesc": "Default: -H Authorization ${token_type} ${access_token}",
        "propertyType": "TEXT",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "OAUTH_ONE_TOKEN_BASED",
    "securityProperties": [
        {
            "displayName": "Consumer Key",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to identify the client that is making the request.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth_consumer_key",
            "propertyShortDesc": "Registered consumer key.",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "Consumer Secret",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to authorize the client that is making the request.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth_consumer_secret",
            "propertyShortDesc": "Registered consumer Secret.",
            "propertyType": "PASSWORD",
            "requiredFlag": true
        }
    ]
}

```



```

    },
    {
      "displayName": "Token",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to access protected resource.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth_token",
      "propertyShortDesc": "Registered token",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Token Secret",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used for generating signature for the request.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth_token_secret",
      "propertyShortDesc": "Token secret provided by server.",
      "propertyType": "PASSWORD",
      "requiredFlag": true
    },
    {
      "displayName": "Realm",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used for identifying the account.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "realm",
      "propertyShortDesc": "Account identifier.",
      "propertyType": "STRING",
      "requiredFlag": false
    },
    {
      "acceptableKeys": [
        {
          "displayName": "HMAC-SHA1",
          "keyName": "HMAC-SHA1",
          "scope": "SOURCE_AND_TARGET"
        },
        {
          "displayName": "HMAC-SHA256",
          "keyName": "HMAC-SHA256",

```

```

        "scope": "SOURCE_AND_TARGET"
    }
],
"defaultPropertyValue": "HMAC-SHA1",
"displayName": "Signature Method (Default Value:HMAC-SHA1)",
"hasAttachment": false,
"hiddenFlag": false,
"propertyDescription": "Used for signing the request.",
"propertyGroup": "CREDENTIALS",
"propertyName": "oauth_signature_method",
"propertyShortDesc": "Signature Method used to sign the request.",
"propertyType": "CHOICE",
"requiredFlag": false
}
]
},
{
    "scope": "TARGET",
    "securityPolicy": "OAUTH_RESOURCE_OWNER_PASSWORD_CREDENTIALS",
    "securityProperties": [
        {
            "displayName": "Access Token URI",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.access.token.uri",
            "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "Client Id",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.client.id",
            "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
            "propertyType": "STRING",
            "requiredFlag": true
        }
    ],
    {

```

```

        "displayName": "Client Secret",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.client.secret",
        "propertyShortDesc": "<unique random string matches your API console project>",
        "propertyType": "PASSWORD",
        "requiredFlag": true
    },
    {
        "displayName": "Scope",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Permissions your application is requesting on behalf of the user.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.scope",
        "propertyShortDesc": "for example: read,write.",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Auth Request Media Type",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Auth Content-Type",
        "propertyGroup": "OTHER",
        "propertyName": "oauth.request.content.type",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Username",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "A username credential",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "username",
        "propertyShortDesc": "Use UserName",
        "propertyType": "STRING",
        "requiredFlag": true
    }
]

```

```

        "displayName": "Password",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "A password credential",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "password",
        "propertyShortDesc": "Enter Password",
        "propertyType": "PASSWORD",
        "requiredFlag": true
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Send client credentials as basic auth header",
                "keyName": "client_credentials_as_header",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "Send client credentials in body",
                "keyName": "client_credentials_in_body",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "displayName": "Client Authentication",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Client Authentication",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "clientAuthentication",
        "propertyShortDesc": "Client Authentication",
        "propertyType": "CHOICE",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "CUSTOM_TWO_LEGGED",
    "securityProperties": [
        {
            "displayName": "Access Token Request",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Access Token Request that should be used to fetch the access token.",

```

```

    "propertyGroup": "CREDENTIALS",
    "propertyName": "accessTokenRequest",
    "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <access-token-uri>?<query params>",
    "propertyType": "TEXT",
    "requiredFlag": true
  },
  {
    "displayName": "Refresh Token Request",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Refresh Token Request that should be used to fetch the access token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "refreshTokenRequest",
    "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <refresh-token-uri>?<query params>",
    "propertyType": "TEXT",
    "requiredFlag": false
  },
  {
    "displayName": "$access_token",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Regex that identifies the access token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$access_token",
    "propertyShortDesc": "Default: access.[tT]oken",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$refresh_token",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Regex that identifies the refresh token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$refresh_token",
    "propertyShortDesc": "Default: refresh.[tT]oken",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$expiry",
    "hasAttachment": false,
    "hiddenFlag": false,

```

```

        "propertyDescription": "Numeric value (in seconds)that specifies the expiry interval or a regex that
identifies when the access token expires.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "$expiry",
        "propertyShortDesc": "Default: expires_in",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "$token_type",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Regex that identifies the access token type.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "$token_type",
        "propertyShortDesc": "Default: token.?[tT]type",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "access_token_usage",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Access token usage. A curl type syntax to illustrate how access token should be passed
to access a protected resource.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "accessTokenUsage",
        "propertyShortDesc": "Default: -H Authorization ${token_type} ${access_token}",
        "propertyType": "TEXT",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "GENERIC_OAUTH_AUTHORIZATION_CODE",
    "securityProperties": [
        {
            "displayName": "Client Id",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
            "propertyGroup": "CREDENTIALS",

```

```

    "propertyName": "oauth.client.id",
    "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Client Secret",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.client.secret",
    "propertyShortDesc": "<unique random string matches your API console project>",
    "propertyType": "PASSWORD",
    "requiredFlag": true
  },
  {
    "displayName": "Authorization Code URI",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "This endpoint is the target of the initial request. It handles active session lookup,
authenticating the user, and user consent.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.auth.code.uri",
    "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/auth",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Access Token URI",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.access.token.uri",
    "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Scope",
    "hasAttachment": false,
    "hiddenFlag": false,

```

```

        "propertyDescription": "Permissions your application is requesting on behalf of the user.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.scope",
        "propertyShortDesc": "for example: read,write.",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Send client credentials in body",
                "keyName": "client_credentials_in_body",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "Send client credentials as basic auth header",
                "keyName": "client_credentials_as_header",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "displayName": "Client Authentication",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Client Authentication",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "clientAuthentication",
        "propertyShortDesc": "Client Authentication",
        "propertyType": "CHOICE",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "JWT_CLIENT_CREDENTIALS_ASSERTION_FOR_OAUTH",
    "securityProperties": [
        {
            "displayName": "Access Token URI",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.access.token.uri",
            "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",

```



```

    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "JWT headers in json format",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT headers in json format",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_header",
    "propertyShortDesc": "JWT headers in json format",
    "propertyType": "URL_OR_FILE",
    "requiredFlag": true
  },
  {
    "displayName": "JWT payload in json format",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT payload in json format",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_payload",
    "propertyShortDesc": "JWT payload in json format",
    "propertyType": "URL_OR_FILE",
    "requiredFlag": true
  },
  {
    "displayName": "JWT Private Key Alias",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT Private Key Alias",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_signing_key_alias_name",
    "propertyShortDesc": "JWT Private Key Alias",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Scope",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Scope",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "scope",
    "propertyShortDesc": "Scope",

```

```

        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Access Token Request",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Default: -X POST -H 'Content-Type: application/x-www-form-urlencoded' -d
'grant_type=client_credentials&client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-
bearer&client_assertion=${client_assertion}&scope=${scope}' ${access_token_uri}?<query params> . Here scope is optional",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "jwtAccessTokenRequest",
        "propertyShortDesc": "Access Token Request that should be used to fetch the access token.",
        "propertyType": "STRING",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "OAUTH_CLIENT_CREDENTIALS",
    "securityProperties": [
        {
            "displayName": "Access Token URI",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.access.token.uri",
            "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "Client Id",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.client.id",
            "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
            "propertyType": "STRING",
            "requiredFlag": true
        }
    ]
}

```

```

    },
    {
      "displayName": "Client Secret",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.client.secret",
      "propertyShortDesc": "<unique random string matches your API console project>",
      "propertyType": "PASSWORD",
      "requiredFlag": true
    },
    {
      "displayName": "Scope",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Permissions your application is requesting on behalf of the user.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.scope",
      "propertyShortDesc": "for example: read,write.",
      "propertyType": "STRING",
      "requiredFlag": false
    },
    {
      "displayName": "Auth Request Media Type",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "application/json or application/x-www-form-urlencoded",
      "propertyGroup": "OTHER",
      "propertyName": "oauth.request.content.type",
      "propertyType": "STRING",
      "requiredFlag": false
    },
    {
      "acceptableKeys": [
        {
          "displayName": "Send client credentials as basic auth header",
          "keyName": "client_credentials_as_header",
          "scope": "SOURCE_AND_TARGET"
        },
        {
          "displayName": "Send client credentials in body",
          "keyName": "client_credentials_in_body",

```

```

        "scope": "SOURCE_AND_TARGET"
    }
],
    "displayName": "Client Authentication",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Client Authentication",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "clientAuthentication",
    "propertyShortDesc": "Client Authentication",
    "propertyType": "CHOICE",
    "requiredFlag": false
}
],
},
{
    "scope": "TARGET",
    "securityPolicy": "AWS_SIGNATURE_VERSION4",
    "securityProperties": [
        {
            "displayName": "Access Key",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyGroup": "CREDENTIALS",
            "propertyName": "AccessKey",
            "propertyShortDesc": "Access Key of AWS",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "Secret Key",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Secret Key of AWS",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "SecretKey",
            "propertyShortDesc": "Secret Key of AWS",
            "propertyType": "PASSWORD",
            "requiredFlag": true
        }
    ],
    {
        "acceptableKeys": [
            {
                "displayName": "US West (Oregon) Region (us-west-2)",

```

```

        "keyName": "us-west-2",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "US West (N. California) Region (us-west-1)",
        "keyName": "us-west-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "US East (Ohio) Region (us-east-2)",
        "keyName": "us-east-2",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "US East (N. Virginia) Region (us-east-1)",
        "keyName": "us-east-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Asia Pacific (Mumbai) Region (ap-south-1)",
        "keyName": "ap-south-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Asia Pacific (Seoul) Region (ap-northeast-2)",
        "keyName": "ap-northeast-2",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Asia Pacific (Tokyo) Region (ap-northeast-1)",
        "keyName": "ap-northeast-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Asia Pacific (Sydney) Region (ap-southeast-2)",
        "keyName": "ap-southeast-2",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Asia Pacific (Singapore) Region (ap-southeast-1)",
        "keyName": "ap-southeast-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {

```

```
    "displayName": "Canada (Central) Region (ca-central-1)",
    "keyName": "ca-central-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "China (Beijing) Region (cn-north-1)",
    "keyName": "cn-north-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "EU (Frankfurt) Region (eu-central-1)",
    "keyName": "eu-central-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "EU (Ireland) Region (eu-west-1)",
    "keyName": "eu-west-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "EU (London) Region (eu-west-2)",
    "keyName": "eu-west-2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "EU (Paris) Region (eu-west-3)",
    "keyName": "eu-west-3",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "South America (São Paulo) Region (sa-east-1)",
    "keyName": "sa-east-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS GovCloud (US) (us-gov-west-1)",
    "keyName": "us-gov-west-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Africa (Cape Town) Region (af-south-1)",
    "keyName": "af-south-1",
    "scope": "SOURCE_AND_TARGET"
  },
}
```

```

{
  "displayName": "Asia Pacific (Hong Kong) Region (ap-east-1)",
  "keyName": "ap-east-1",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Asia Pacific (Hyderabad) Region (ap-south-2)",
  "keyName": "ap-south-2",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Asia Pacific (Jakarta) Region (ap-southeast-3)",
  "keyName": "ap-southeast-3",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Asia Pacific (Melbourne) Region (ap-southeast-4)",
  "keyName": "ap-southeast-4",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Asia Pacific (Osaka) Region (ap-northeast-3)",
  "keyName": "ap-northeast-3",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Europe (Milan) Region (eu-south-1)",
  "keyName": "eu-south-1",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Europe (Spain) Region (eu-south-2)",
  "keyName": "eu-south-2",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Europe (Stockholm) Region (eu-north-1)",
  "keyName": "eu-north-1",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Europe (Zurich) Region (eu-central-2)",
  "keyName": "eu-central-2",
  "scope": "SOURCE_AND_TARGET"
}

```

```

    },
    {
      "displayName": "Israel (Tel Aviv) Region (il-central-1)",
      "keyName": "il-central-1",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Middle East (Bahrain) Region (me-south-1)",
      "keyName": "me-south-1",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Middle East (UAE) (me-central-1)",
      "keyName": "me-central-1",
      "scope": "SOURCE_AND_TARGET"
    }
  ],
  "displayName": "AWS Region",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "AWS Region",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "AWSRegion",
  "propertyShortDesc": "AWS Region",
  "propertyType": "CHOICE",
  "requiredFlag": true
},
{
  "acceptableKeys": [
    {
      "displayName": "Alexa for Business",
      "keyName": "a4b",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon API Gateway",
      "keyName": "apigateway",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon AppStream 2.0",
      "keyName": "appstream2",
      "scope": "SOURCE_AND_TARGET"
    }
  ],

```



```
{
  "displayName": "AWS AppSync Control Plane",
  "keyName": "appsync",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS AppSync Data Plane",
  "keyName": "appsync-api",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Athena",
  "keyName": "athena",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Auto Scaling",
  "keyName": "autoscaling",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Batch",
  "keyName": "batch",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Cost Explorer",
  "keyName": "ce",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Cost and Usage Reports",
  "keyName": "cur",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Budgets",
  "keyName": "budgets",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Price List Service",
  "keyName": "api.pricing",
  "scope": "SOURCE_AND_TARGET"
}
```

```

    },
    {
      "displayName": "AWS Certificate Manager",
      "keyName": "acm",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Certificate Manager Private Certificate Authority",
      "keyName": "acm-pca",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Cloud9",
      "keyName": "cloud9",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Cloud Directory",
      "keyName": "clouddirectory",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS CloudFormation",
      "keyName": "cloudformation",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon CloudFront",
      "keyName": "cloudfront",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS CloudHSM",
      "keyName": "cloudhsmv2",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS CloudHSM Classic",
      "keyName": "cloudhsm",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon CloudSearch",
      "keyName": "cloudsearch",

```

```
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CloudTrail",
    "keyName": "cloudtrail",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon CloudWatch",
    "keyName": "monitoring",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon CloudWatch Events",
    "keyName": "events",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Execute API",
    "keyName": "execute-api",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon CloudWatch Logs",
    "keyName": "logs",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CodeBuild",
    "keyName": "codebuild",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CodeCommit",
    "keyName": "codecommit",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CodeDeploy",
    "keyName": "codedeploy",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CodePipeline",
```

```

        "keyName": "codepipeline",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS CodeStar",
        "keyName": "codestar",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon Cognito Your User Pools",
        "keyName": "cognito-idp",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon Cognito Federated Identities",
        "keyName": "cognito-identity",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon Cognito Sync",
        "keyName": "cognito-sync",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon Comprehend",
        "keyName": "Comprehend",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Config",
        "keyName": "config",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Config Rules",
        "keyName": "configrules",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon Connect",
        "keyName": "connect",
        "scope": "SOURCE_AND_TARGET"
    },
    {

```

```
    "displayName": "AWS Data Pipeline",
    "keyName": "datapipeline",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Database Migration Service",
    "keyName": "dms",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS DeepLens",
    "keyName": "deeplens",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Device Farm",
    "keyName": "devicefarm",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Direct Connect",
    "keyName": "directconnect",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Directory Service",
    "keyName": "ds",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon DynamoDB",
    "keyName": "dynamodb",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "DynamoDB Accelerator (DAX)",
    "keyName": "dax",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon DynamoDB Streams",
    "keyName": "streams.dynamodb",
    "scope": "SOURCE_AND_TARGET"
  },
}
```

```
{
  "displayName": "AWS Elastic Beanstalk",
  "keyName": "elasticbeanstalk",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Elastic Beanstalk Health Service",
  "keyName": "elasticbeanstalk-health",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Elastic Compute Cloud (Amazon EC2)",
  "keyName": "ec2",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Elastic Container Service",
  "keyName": "ecr",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Elastic Container Service",
  "keyName": "ecs",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Elastic Container Service for Kubernetes (Amazon EKS)",
  "keyName": "eks",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Elastic File System",
  "keyName": "elasticfilesystem",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Elastic Load Balancing",
  "keyName": "elasticloadbalancing",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Elastic Transcoder",
  "keyName": "elastictranscoder",
  "scope": "SOURCE_AND_TARGET"
}
```

```
    },  
    {  
      "displayName": "Amazon ElastiCache",  
      "keyName": "elasticache",  
      "scope": "SOURCE_AND_TARGET"  
    },  
    {  
      "displayName": "Amazon Elasticsearch Service",  
      "keyName": "es",  
      "scope": "SOURCE_AND_TARGET"  
    },  
    {  
      "displayName": "Amazon EMR",  
      "keyName": "elasticmapreduce",  
      "scope": "SOURCE_AND_TARGET"  
    },  
    {  
      "displayName": "AWS Firewall Manager",  
      "keyName": "fms",  
      "scope": "SOURCE_AND_TARGET"  
    },  
    {  
      "displayName": "Amazon GameLift",  
      "keyName": "gamelift",  
      "scope": "SOURCE_AND_TARGET"  
    },  
    {  
      "displayName": "Amazon Glacier",  
      "keyName": "glacier",  
      "scope": "SOURCE_AND_TARGET"  
    },  
    {  
      "displayName": "AWS Glue",  
      "keyName": "glue",  
      "scope": "SOURCE_AND_TARGET"  
    },  
    {  
      "displayName": "AWS Greengrass",  
      "keyName": "greengrass",  
      "scope": "SOURCE_AND_TARGET"  
    },  
    {  
      "displayName": "Amazon GuardDuty",  
      "keyName": "guardduty",
```

```

    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Health",
    "keyName": "health",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Identity and Access Management (IAM)",
    "keyName": "iam",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Import/Export",
    "keyName": "importexport",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Inspector",
    "keyName": "inspector",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS IoT 1-Click",
    "keyName": "iot1click",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS IoT Analytics",
    "keyName": "iotanalytics",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS IoT Core",
    "keyName": "iot",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Key Management Service",
    "keyName": "kms",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Kinesis Data Analytics",

```



```
    "keyName": "kinesisanalytics",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Kinesis Data Firehose",
    "keyName": "firehose",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Kinesis Data Streams",
    "keyName": "kinesis",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Kinesis Video Streams",
    "keyName": "kinesisvideo",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Lambda",
    "keyName": "lambda",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Lex (Model Building Endpoints)",
    "keyName": "models.lex",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Lex (Runtime Endpoints)",
    "keyName": "runtime.lex",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Lightsail",
    "keyName": "lightsail",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Macie",
    "keyName": "macie",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    
```

```
    "displayName": "Amazon Machine Learning",
    "keyName": "machinelearning",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaConvert",
    "keyName": "mediaconvert",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaLive",
    "keyName": "medialive",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaPackage",
    "keyName": "mediapackage",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaStore",
    "keyName": "mediastore",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaTailor",
    "keyName": "mediatailor",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Migration Hub",
    "keyName": "mgh",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Mobile Analytics",
    "keyName": "mobileanalytics",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon MQ",
    "keyName": "mq",
    "scope": "SOURCE_AND_TARGET"
  },
}
```

```
{
  "displayName": "AWS OpsWorks (CM)",
  "keyName": "opsworks-cm",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS OpsWorks (Stacks)",
  "keyName": "opsworks",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Organizations",
  "keyName": "organizations",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Pinpoint",
  "keyName": "pinpoint",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Polly",
  "keyName": "polly",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon QuickSight",
  "keyName": "quicksight",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Redshift",
  "keyName": "redshift",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Rekognition",
  "keyName": "rekognition",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Relational Database Service (Amazon RDS)",
  "keyName": "rds",
  "scope": "SOURCE_AND_TARGET"
}
```

```

    },
    {
      "displayName": "AWS Resource Groups",
      "keyName": "resource-groups",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Secrets Manager",
      "keyName": "secretsmanager",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Server Migration Service",
      "keyName": "sms",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon SageMaker",
      "keyName": "api.sagemaker",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Security Token Service (AWS STS)",
      "keyName": "sts",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Serverless Application Repository",
      "keyName": "serverlessrepo",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Service Catalog",
      "keyName": "servicecatalog",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Shield Advanced",
      "keyName": "Shield",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Simple Email Service (Amazon SES)",
      "keyName": "email-smtp",

```

```

    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Simple Notification Service (Amazon SNS)",
    "keyName": "sns",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Simple Queue Service (Amazon SQS)",
    "keyName": "sqs",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Simple Storage Service (Amazon S3)",
    "keyName": "s3",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Simple Storage Service Website Endpoints",
    "keyName": "s3-website",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Simple Workflow Service (Amazon SWF)",
    "keyName": "swf",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon SimpleDB",
    "keyName": "sdb",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Snowball",
    "keyName": "snowball",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Step Functions",
    "keyName": "states",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Storage Gateway",

```

```

        "keyName": "storagegateway",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Systems Manager",
        "keyName": "ssm",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon Transcribe",
        "keyName": "transcribe",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon Translate",
        "keyName": "Translate",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon Translate (FIPS)",
        "keyName": "translate-fips",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon VPC",
        "keyName": "waf",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon WorkDocs",
        "keyName": "workdocs",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Amazon WorkSpaces",
        "keyName": "workspaces",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS X-Ray",
        "keyName": "xray",
        "scope": "SOURCE_AND_TARGET"
    }
],

```

```

        "displayName": "Service Name",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Service Name like S3",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "ServiceName",
        "propertyShortDesc": "Service Name",
        "propertyType": "CHOICE",
        "requiredFlag": true
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "NONE",
    "securityProperties": []
},
{
    "scope": "SOURCE_AND_TARGET",
    "securityPolicy": "BASIC_AUTH",
    "securityProperties": [
        {
            "displayName": "Username",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A username credential",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "username",
            "propertyShortDesc": "Use UserName",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "Password",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A password credential",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "password",
            "propertyShortDesc": "Enter Password",
            "propertyType": "PASSWORD",
            "requiredFlag": true
        }
    ]
}
]

```

```

    }
  ],
  "securityProperties": [
    {
      "displayName": "Access Token URI",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.access.token.uri",
      "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
      "propertyType": "STRING",
      "propertyValue": "https://idcs-5ba32fa3496f48289532f8fc10f47032.identity.oraclecloud.com:443/oauth2/v1/token",
      "requiredFlag": true
    },
    {
      "displayName": "Client Id",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to identify the client(the software requesting an access token) that is making the
request. The value passed in this parameter must exactly match the value shown in your API console project.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.client.id",
      "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
      "propertyType": "STRING",
      "propertyValue": "<<your-IntegratedApp-ClientID>>",
      "requiredFlag": true
    },
    {
      "displayName": "Client Secret",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making the
request. The value passed in this parameter must exactly match the value shown in your API console project.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.client.secret",
      "propertyShortDesc": "<unique random string matches your API console project>",
      "propertyType": "PASSWORD",
      "propertyValue": "*****",
      "requiredFlag": true
    },
    {
      "displayName": "Scope",
      "hasAttachment": false,

```



```

        "hiddenFlag": false,
        "propertyDescription": "Permissions your application is requesting on behalf of the user.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.scope",
        "propertyShortDesc": "for example: read,write.",
        "propertyType": "STRING",
        "propertyValue": "",
        "requiredFlag": false
    },
    {
        "displayName": "Auth Request Media Type",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Auth Content-Type",
        "propertyGroup": "OTHER",
        "propertyName": "oauth.request.content.type",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Username",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "A username credential",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "username",
        "propertyShortDesc": "Use UserName",
        "propertyType": "STRING",
        "propertyValue": "peter.obert+local@oracle.com",
        "requiredFlag": true
    },
    {
        "displayName": "Password",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "A password credential",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "password",
        "propertyShortDesc": "Enter Password",
        "propertyType": "PASSWORD",
        "propertyValue": "*****",
        "requiredFlag": true
    },
    {

```

```

    "acceptableKeys": [
      {
        "displayName": "Send client credentials as basic auth header",
        "keyName": "client_credentials_as_header",
        "scope": "SOURCE_AND_TARGET"
      },
      {
        "displayName": "Send client credentials in body",
        "keyName": "client_credentials_in_body",
        "scope": "SOURCE_AND_TARGET"
      }
    ],
    "displayName": "Client Authentication",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Client Authentication",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "clientAuthentication",
    "propertyShortDesc": "Client Authentication",
    "propertyType": "CHOICE",
    "propertyValue": "client_credentials_as_header",
    "requiredFlag": false
  }
],
"status": "CONFIGURED",
"testStatus": false,
"usage": 4,
"usageActive": 1
}

```

Example

OIC3_DEV_UPDATEconnectionTESTNETSUITE

Request

[View More](#)

cURL

```

curl --location 'https://design.integration.<<your-OIC-Instance-
region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/?integrationInstance=%3C%3Cyour-OIC-InstanceName%3E%3E' \
--header 'X-HTTP-Method-Override: PATCH' \
--data-raw '{
  "adapterType": {
    "displayName": "REST",
    "links": [],
    "name": "rest",

```

```
    "type": "PREINSTALLED"
  },
  "agentRequired": false,
  "agentSupported": true,
  "connectionProperties": [
    {
      "acceptableKeys": [
        {
          "displayName": "REST API Base URL",
          "keyName": "restUrl",
          "scope": "SOURCE_AND_TARGET"
        },
        {
          "displayName": "Open API (1.0/2.0/3.0) URL",
          "keyName": "openApiUrl",
          "scope": "SOURCE_AND_TARGET"
        }
      ],
      "displayName": "Connection Type",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyGroup": "CONNECTION_PROPS",
      "propertyName": "connectionType",
      "propertyType": "CHOICE",
      "propertyValue": "restUrl",
      "requiredFlag": true
    },
    {
      "acceptableKeys": [
        {
          "displayName": "TLSv1.1",
          "keyName": "TLSv1.1",
          "scope": "SOURCE_AND_TARGET"
        },
        {
          "displayName": "TLSv1.2",
          "keyName": "TLSv1.2",
          "scope": "SOURCE_AND_TARGET"
        }
      ],
      "displayName": "TLS Version",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyGroup": "CONNECTION_PROPS",
```

```

        "propertyName": "tlsVersion",
        "propertyShortDesc": "TLS Version",
        "propertyType": "CHOICE",
        "requiredFlag": false
    },
    {
        "displayName": "Connection URL",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "connectionUrl",
        "propertyShortDesc": "Please make sure that this value really corresponds to the type selected above.",
        "propertyType": "URL",
        "propertyValue": "https://opa-oic-teamoic3-1740047240850-<<Storage-namespace>>-fra.process.oc1.oraclecloud.com/process/api",
        "requiredFlag": true
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Yes",
                "keyName": "true",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "No",
                "keyName": "false",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "displayName": "Enable two way SSL for outbound connections (Optional)",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Both the client and the server pass certificates to each other to establish an SSL link, when
two-way SSL is enabled. Both sides now know the identity of the other from their respective certificates.",
        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "enableTwoWaySSL",
        "propertyShortDesc": "Select \"Yes\" if both the client and server should authenticate each other.",
        "propertyType": "CHOICE",
        "requiredFlag": false
    },
    {
        "displayName": "Identity keystore alias name (Optional)",
        "hasAttachment": false,

```

```

        "hiddenFlag": false,
        "propertyDescription": "Both the client and the server pass certificates to each other to establish an SSL link, when
two-way SSL is enabled. Name of the identity certificate alias to use for that communication. This value should match the alias
that was provided to import identity to ICS.",
        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "sslCertificateAlias",
        "propertyShortDesc": "Name of alias to use for establishing identity during two way SSL communication",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    ],
    "created": "2023-11-13T15:58:08.535+0000",
    "createdBy": "peter.obert@oracle.com",
    "description": "Invoking the process automation to create new process instances",
    "hasOverride": false,
    "id": "POB_OICTOOPA_INVOKE",
    "keywords": "",
    "lastRefreshedDate": "2025-02-21T06:51:49.587+0000",
    "lastUpdated": "2025-03-07T13:40:44.908+0000",
    "lastUpdatedBy": "peter.obert@oracle.com",
    "links": [
        {
            "href": "https://design.integration.<<your-OIC-Instance-
region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/POB_OICTOOPA_INVOKE?integrationInstance=teamoic3-<<Storage-
namespace>>-fr",
            "rel": "self"
        },
        {
            "href": "https://design.integration.<<your-OIC-Instance-
region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/POB_OICTOOPA_INVOKE?integrationInstance=teamoic3-<<Storage-
namespace>>-fr",
            "rel": "canonical"
        }
    ],
    "lockedFlag": false,
    "metadataDownloadState": "COMPLETE",
    "metadataDownloadSupportedFlag": true,
    "name": "pob-OICtoOPA-Invoke",
    "percentageComplete": 100,
    "privateEndpoint": false,
    "privateEndpointSupported": true,
    "refreshedStatus": "COMPLETE",
    "role": "TARGET",
    "securityPolicy": "OAUTH_RESOURCE_OWNER_PASSWORD_CREDENTIALS",

```

```

"securityPolicyInfo": [
  {
    "scope": "TARGET",
    "securityPolicy": "OCI_SERVICE_INVOCATION",
    "securityProperties": [
      {
        "defaultPropertyValue": "DefaultValueForRPST",
        "hasAttachment": false,
        "hiddenFlag": true,
        "propertyGroup": "CREDENTIALS",
        "propertyName": "dummy_credential_rpst",
        "propertyType": "STRING",
        "requiredFlag": true
      }
    ]
  },
  {
    "scope": "TARGET",
    "securityPolicy": "OCI_SIGNATURE_VERSION1",
    "securityProperties": [
      {
        "displayName": "Tenancy OCID",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyGroup": "CREDENTIALS",
        "propertyName": "TenancyOCID",
        "propertyShortDesc": "Tenancy OCID of OCI",
        "propertyType": "STRING",
        "requiredFlag": true
      },
      {
        "displayName": "User OCID",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "User OCID of OCI",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "UserOCID",
        "propertyShortDesc": "User OCID of OCI",
        "propertyType": "STRING",
        "requiredFlag": true
      },
      {
        "displayName": "Private Key",
        "hasAttachment": false,

```

```

        "hiddenFlag": false,
        "propertyDescription": "OCI Private Key",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "PrivateKey",
        "propertyShortDesc": "OCI Private Key",
        "propertyType": "URL_OR_FILE",
        "requiredFlag": true
    },
    {
        "displayName": "Finger Print",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Finger Print",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "FingerPrint",
        "propertyShortDesc": "Finger Print",
        "propertyType": "STRING",
        "requiredFlag": true
    },
    {
        "displayName": "Pass Phrase",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Password entered at the time of generating the public/private key.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "PassPhrase",
        "propertyShortDesc": "Password entered at the time of generating the public/private key.",
        "propertyType": "PASSWORD",
        "requiredFlag": false
    }
],
{
    "scope": "TARGET",
    "securityPolicy": "JWT_USER_ASSERTION_FOR_OAUTH",
    "securityProperties": [
        {
            "displayName": "Access Token URI",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.access.token.uri",
            "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",

```

```

    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "JWT headers in json format",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT headers in json format",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_header",
    "propertyShortDesc": "JWT headers in json format",
    "propertyType": "URL_OR_FILE",
    "requiredFlag": true
  },
  {
    "displayName": "JWT payload in json format",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT payload in json format",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_payload",
    "propertyShortDesc": "JWT payload in json format",
    "propertyType": "URL_OR_FILE",
    "requiredFlag": true
  },
  {
    "displayName": "JWT Private Key Alias",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT Private Key Alias",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_signing_key_alias_name",
    "propertyShortDesc": "JWT Private Key Alias",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Scope",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Scope",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "scope",
    "propertyShortDesc": "Scope",

```



```

        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Access Token Request",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Default: -X POST -H ''Content-Type: application/x-www-form-urlencoded'' -d
''grant_type=urn:ietf:params:oauth:grant-type:jwt-
bearer&assertion=${user_assertion}&client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-
bearer&client_assertion=${client_assertion}&scope=${scope}'' ${access_token_uri}?<query params> . Here scope is optional",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "jwtAccessTokenRequest",
        "propertyShortDesc": "Access Token Request that should be used to fetch the access token.",
        "propertyType": "STRING",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "CUSTOM_SINGLE_TOKEN",
    "securityProperties": [
        {
            "displayName": "API Key",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Generated API Key used to identify the client that is making the request.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "accessToken",
            "propertyShortDesc": "Generated API key.",
            "propertyType": "PASSWORD",
            "requiredFlag": true
        },
        {
            "displayName": "API Key Usage",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "API Key Usage. URI syntax to illustrate how the API KEY should be passed to access a
protected resource. Example: To pass the API-KEY as a query parameter \"?<query-param-name>=${api-key}\" or To pass the API-KEY as
a Header \"-H Authorization: Bearer ${api-key}\" without quotes.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "accessTokenUsage",
            "propertyShortDesc": "Default: -H Authorization: Bearer ${api-key}",
        }
    ]
}

```

```

        "propertyType": "TEXT",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "OAUTH_AUTHORIZATION_CODE_CREDENTIALS",
    "securityProperties": [
        {
            "displayName": "Client Id",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.client.id",
            "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "Client Secret",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.client.secret",
            "propertyShortDesc": "<unique random string matches your API console project>",
            "propertyType": "PASSWORD",
            "requiredFlag": true
        },
        {
            "displayName": "Authorization Code URI",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "This endpoint is the target of the initial request. It handles active session lookup,
authenticating the user, and user consent.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.auth.code.uri",
            "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/auth",
            "propertyType": "STRING",
            "requiredFlag": true
        }
    ]
}

```

```

    },
    {
      "displayName": "Access Token URI",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.access.token.uri",
      "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Scope",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Permissions your application is requesting on behalf of the user.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.scope",
      "propertyShortDesc": "for example: read,write.",
      "propertyType": "STRING",
      "requiredFlag": false
    },
    {
      "acceptableKeys": [
        {
          "displayName": "Send client credentials in body",
          "keyName": "client_credentials_in_body",
          "scope": "SOURCE_AND_TARGET"
        },
        {
          "displayName": "Send client credentials as basic auth header",
          "keyName": "client_credentials_as_header",
          "scope": "SOURCE_AND_TARGET"
        }
      ],
      "displayName": "Client Authentication",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Client Authentication",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "clientAuthentication",
      "propertyShortDesc": "Client Authentication",
      "propertyType": "CHOICE",

```

```

        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "CUSTOM_THREE_LEGGED",
    "securityProperties": [
        {
            "displayName": "Authorization Request",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Authorization url where the user will be redirected to provide consent.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "authRequest",
            "propertyShortDesc": "Example : <authUri>?<queryParams>",
            "propertyType": "TEXT",
            "requiredFlag": true
        },
        {
            "displayName": "Access Token Request",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Access Token Request that should be used to fetch the access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "accessTokenRequest",
            "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <access-token-uri>?<query params>",
            "propertyType": "TEXT",
            "requiredFlag": true
        },
        {
            "displayName": "Refresh Token Request",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Refresh Token Request that should be used to fetch the access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "refreshTokenRequest",
            "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <refresh-token-uri>?<query params>",
            "propertyType": "TEXT",
            "requiredFlag": false
        },
        {
            "displayName": "$auth_code",
            "hasAttachment": false,

```

```

    "hiddenFlag": false,
    "propertyDescription": "Regex that identifies the authorization code.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$auth_code",
    "propertyShortDesc": "Default: code",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$access_token",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Regex that identifies the access token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$access_token",
    "propertyShortDesc": "Default: access.[tT]oken",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$refresh_token",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Regex that identifies the refresh token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$refresh_token",
    "propertyShortDesc": "Default: refresh.[tT]oken",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$expiry",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Numeric value (in seconds)that specifies the expiry interval or a regex that
identifies when the access token expires.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$expiry",
    "propertyShortDesc": "Default: expires_in",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$token_type",

```

```

        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Regex that identifies the access token type.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "$token_type",
        "propertyShortDesc": "Default: token.?[tT]ype",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "access_token_usage",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Access token usage. A curl type syntax to illustrate how access token should be passed
to access a protected resource.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "accessTokenUsage",
        "propertyShortDesc": "Default: -H Authorization ${token_type} ${access_token}",
        "propertyType": "TEXT",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "OAUTH_ONE_TOKEN_BASED",
    "securityProperties": [
        {
            "displayName": "Consumer Key",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to identify the client that is making the request.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth_consumer_key",
            "propertyShortDesc": "Registered consumer key.",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "Consumer Secret",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to authorize the client that is making the request.",
            "propertyGroup": "CREDENTIALS",

```

```

    "propertyName": "oauth_consumer_secret",
    "propertyShortDesc": "Registered consumer Secret.",
    "propertyType": "PASSWORD",
    "requiredFlag": true
  },
  {
    "displayName": "Token",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Used to access protected resource.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth_token",
    "propertyShortDesc": "Registered token",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Token Secret",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Used for generating signature for the request.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth_token_secret",
    "propertyShortDesc": "Token secret provided by server.",
    "propertyType": "PASSWORD",
    "requiredFlag": true
  },
  {
    "displayName": "Realm",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Used for identifying the account.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "realm",
    "propertyShortDesc": "Account identifier.",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "acceptableKeys": [
      {
        "displayName": "HMAC-SHA1",
        "keyName": "HMAC-SHA1",
        "scope": "SOURCE_AND_TARGET"
      }
    ]
  }

```

```

    },
    {
      "displayName": "HMAC-SHA256",
      "keyName": "HMAC-SHA256",
      "scope": "SOURCE_AND_TARGET"
    }
  ],
  "defaultPropertyValue": "HMAC-SHA1",
  "displayName": "Signature Method (Default Value:HMAC-SHA1)",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "Used for signing the request.",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "oauth_signature_method",
  "propertyShortDesc": "Signature Method used to sign the request.",
  "propertyType": "CHOICE",
  "requiredFlag": false
}
]
},
{
  "scope": "TARGET",
  "securityPolicy": "OAUTH_RESOURCE_OWNER_PASSWORD_CREDENTIALS",
  "securityProperties": [
    {
      "displayName": "Access Token URI",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.access.token.uri",
      "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Client Id",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "oauth.client.id",
      "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",

```



```

    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Client Secret",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.client.secret",
    "propertyShortDesc": "<unique random string matches your API console project>",
    "propertyType": "PASSWORD",
    "requiredFlag": true
  },
  {
    "displayName": "Scope",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Permissions your application is requesting on behalf of the user.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.scope",
    "propertyShortDesc": "for example: read,write.",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "Auth Request Media Type",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Auth Content-Type",
    "propertyGroup": "OTHER",
    "propertyName": "oauth.request.content.type",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "Username",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "A username credential",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "username",
    "propertyShortDesc": "Use UserName",

```

```

        "propertyType": "STRING",
        "requiredFlag": true
    },
    {
        "displayName": "Password",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "A password credential",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "password",
        "propertyShortDesc": "Enter Password",
        "propertyType": "PASSWORD",
        "requiredFlag": true
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Send client credentials as basic auth header",
                "keyName": "client_credentials_as_header",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "Send client credentials in body",
                "keyName": "client_credentials_in_body",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "displayName": "Client Authentication",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Client Authentication",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "clientAuthentication",
        "propertyShortDesc": "Client Authentication",
        "propertyType": "CHOICE",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "CUSTOM_TWO_LEGGED",
    "securityProperties": [
        {

```

```

    "displayName": "Access Token Request",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Access Token Request that should be used to fetch the access token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "accessTokenRequest",
    "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <access-token-uri>?<query params>",
    "propertyType": "TEXT",
    "requiredFlag": true
  },
  {
    "displayName": "Refresh Token Request",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Refresh Token Request that should be used to fetch the access token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "refreshTokenRequest",
    "propertyShortDesc": "Example: -X <Method> -H <headers> -d <string-data> <refresh-token-uri>?<query params>",
    "propertyType": "TEXT",
    "requiredFlag": false
  },
  {
    "displayName": "$access_token",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Regex that identifies the access token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$access_token",
    "propertyShortDesc": "Default: access.[tT]oken",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {
    "displayName": "$refresh_token",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Regex that identifies the refresh token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "$refresh_token",
    "propertyShortDesc": "Default: refresh.[tT]oken",
    "propertyType": "STRING",
    "requiredFlag": false
  },
  {

```

```

        "displayName": "$expiry",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Numeric value (in seconds)that specifies the expiry interval or a regex that
identifies when the access token expires.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "$expiry",
        "propertyShortDesc": "Default: expires_in",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "$token_type",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Regex that identifies the access token type.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "$token_type",
        "propertyShortDesc": "Default: token.?[tT]ype",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "access_token_usage",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Access token usage. A curl type syntax to illustrate how access token should be passed
to access a protected resource.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "accessTokenUsage",
        "propertyShortDesc": "Default: -H Authorization ${token_type} ${access_token}",
        "propertyType": "TEXT",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "GENERIC_OAUTH_AUTHORIZATION_CODE",
    "securityProperties": [
        {
            "displayName": "Client Id",
            "hasAttachment": false,
            "hiddenFlag": false,

```

```

        "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.client.id",
        "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
        "propertyType": "STRING",
        "requiredFlag": true
    },
    {
        "displayName": "Client Secret",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.client.secret",
        "propertyShortDesc": "<unique random string matches your API console project>",
        "propertyType": "PASSWORD",
        "requiredFlag": true
    },
    {
        "displayName": "Authorization Code URI",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "This endpoint is the target of the initial request. It handles active session lookup,
authenticating the user, and user consent.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.auth.code.uri",
        "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/auth",
        "propertyType": "STRING",
        "requiredFlag": true
    },
    {
        "displayName": "Access Token URI",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.access.token.uri",
        "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
        "propertyType": "STRING",
        "requiredFlag": true
    },
    {

```

```

        "displayName": "Scope",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Permissions your application is requesting on behalf of the user.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.scope",
        "propertyShortDesc": "for example: read,write.",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Send client credentials in body",
                "keyName": "client_credentials_in_body",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "Send client credentials as basic auth header",
                "keyName": "client_credentials_as_header",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "displayName": "Client Authentication",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Client Authentication",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "clientAuthentication",
        "propertyShortDesc": "Client Authentication",
        "propertyType": "CHOICE",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "JWT_CLIENT_CREDENTIALS_ASSERTION_FOR_OAUTH",
    "securityProperties": [
        {
            "displayName": "Access Token URI",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A request should be sent to this URI for obtaining an access token.",

```

```

    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.access.token.uri",
    "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "JWT headers in json format",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT headers in json format",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_header",
    "propertyShortDesc": "JWT headers in json format",
    "propertyType": "URL_OR_FILE",
    "requiredFlag": true
  },
  {
    "displayName": "JWT payload in json format",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT payload in json format",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_payload",
    "propertyShortDesc": "JWT payload in json format",
    "propertyType": "URL_OR_FILE",
    "requiredFlag": true
  },
  {
    "displayName": "JWT Private Key Alias",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "JWT Private Key Alias",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "jwt_signing_key_alias_name",
    "propertyShortDesc": "JWT Private Key Alias",
    "propertyType": "STRING",
    "requiredFlag": true
  },
  {
    "displayName": "Scope",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Scope",

```

```

        "propertyGroup": "CREDENTIALS",
        "propertyName": "scope",
        "propertyShortDesc": "Scope",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Access Token Request",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Default: -X POST -H ''Content-Type: application/x-www-form-urlencoded'' -d
        ''grant_type=client_credentials&client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-
        bearer&client_assertion=${client_assertion}&scope=${scope}'' ${access_token_uri}<query params> . Here scope is optional",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "jwtAccessTokenRequest",
        "propertyShortDesc": "Access Token Request that should be used to fetch the access token.",
        "propertyType": "STRING",
        "requiredFlag": false
    }
]
},
{
    "scope": "TARGET",
    "securityPolicy": "OAUTH_CLIENT_CREDENTIALS",
    "securityProperties": [
        {
            "displayName": "Access Token URI",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.access.token.uri",
            "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "Client Id",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Used to identify the client(the software requesting an access token) that is making
            the request. The value passed in this parameter must exactly match the value shown in your API console project.",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "oauth.client.id",

```



```

        "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
        "propertyType": "STRING",
        "requiredFlag": true
    },
    {
        "displayName": "Client Secret",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making
the request. The value passed in this parameter must exactly match the value shown in your API console project.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.client.secret",
        "propertyShortDesc": "<unique random string matches your API console project>",
        "propertyType": "PASSWORD",
        "requiredFlag": true
    },
    {
        "displayName": "Scope",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Permissions your application is requesting on behalf of the user.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.scope",
        "propertyShortDesc": "for example: read,write.",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Auth Request Media Type",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "application/json or application/x-www-form-urlencoded",
        "propertyGroup": "OTHER",
        "propertyName": "oauth.request.content.type",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Send client credentials as basic auth header",
                "keyName": "client_credentials_as_header",
                "scope": "SOURCE_AND_TARGET"
            }
        ]
    }

```

```

        {
            "displayName": "Send client credentials in body",
            "keyName": "client_credentials_in_body",
            "scope": "SOURCE_AND_TARGET"
        }
    ],
    "displayName": "Client Authentication",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Client Authentication",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "clientAuthentication",
    "propertyShortDesc": "Client Authentication",
    "propertyType": "CHOICE",
    "requiredFlag": false
}
]
},
{
    "scope": "TARGET",
    "securityPolicy": "AWS_SIGNATURE_VERSION4",
    "securityProperties": [
        {
            "displayName": "Access Key",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyGroup": "CREDENTIALS",
            "propertyName": "AccessKey",
            "propertyShortDesc": "Access Key of AWS",
            "propertyType": "STRING",
            "requiredFlag": true
        },
        {
            "displayName": "Secret Key",
            "hasAttachment": false,
            "hiddenFlag": false,
            "propertyDescription": "Secret Key of AWS",
            "propertyGroup": "CREDENTIALS",
            "propertyName": "SecretKey",
            "propertyShortDesc": "Secret Key of AWS",
            "propertyType": "PASSWORD",
            "requiredFlag": true
        }
    ],
    {

```

```

"acceptableKeys": [
  {
    "displayName": "US West (Oregon) Region (us-west-2)",
    "keyName": "us-west-2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "US West (N. California) Region (us-west-1)",
    "keyName": "us-west-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "US East (Ohio) Region (us-east-2)",
    "keyName": "us-east-2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "US East (N. Virginia) Region (us-east-1)",
    "keyName": "us-east-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Mumbai) Region (ap-south-1)",
    "keyName": "ap-south-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Seoul) Region (ap-northeast-2)",
    "keyName": "ap-northeast-2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Tokyo) Region (ap-northeast-1)",
    "keyName": "ap-northeast-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Sydney) Region (ap-southeast-2)",
    "keyName": "ap-southeast-2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Singapore) Region (ap-southeast-1)",
    "keyName": "ap-southeast-1",

```

```

    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Canada (Central) Region (ca-central-1)",
    "keyName": "ca-central-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "China (Beijing) Region (cn-north-1)",
    "keyName": "cn-north-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "EU (Frankfurt) Region (eu-central-1)",
    "keyName": "eu-central-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "EU (Ireland) Region (eu-west-1)",
    "keyName": "eu-west-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "EU (London) Region (eu-west-2)",
    "keyName": "eu-west-2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "EU (Paris) Region (eu-west-3)",
    "keyName": "eu-west-3",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "South America (São Paulo) Region (sa-east-1)",
    "keyName": "sa-east-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS GovCloud (US) (us-gov-west-1)",
    "keyName": "us-gov-west-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Africa (Cape Town) Region (af-south-1)",

```

```

    "keyName": "af-south-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Hong Kong) Region (ap-east-1)",
    "keyName": "ap-east-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Hyderabad) Region (ap-south-2)",
    "keyName": "ap-south-2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Jakarta) Region (ap-southeast-3)",
    "keyName": "ap-southeast-3",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Melbourne) Region (ap-southeast-4)",
    "keyName": "ap-southeast-4",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Asia Pacific (Osaka) Region (ap-northeast-3)",
    "keyName": "ap-northeast-3",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Europe (Milan) Region (eu-south-1)",
    "keyName": "eu-south-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Europe (Spain) Region (eu-south-2)",
    "keyName": "eu-south-2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Europe (Stockholm) Region (eu-north-1)",
    "keyName": "eu-north-1",
    "scope": "SOURCE_AND_TARGET"
  },
  {

```

```

        "displayName": "Europe (Zurich) Region (eu-central-2)",
        "keyName": "eu-central-2",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Israel (Tel Aviv) Region (il-central-1)",
        "keyName": "il-central-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Middle East (Bahrain) Region (me-south-1)",
        "keyName": "me-south-1",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Middle East (UAE) (me-central-1)",
        "keyName": "me-central-1",
        "scope": "SOURCE_AND_TARGET"
    }
],
"displayName": "AWS Region",
"hasAttachment": false,
"hiddenFlag": false,
"propertyDescription": "AWS Region",
"propertyGroup": "CREDENTIALS",
"propertyName": "AWSRegion",
"propertyShortDesc": "AWS Region",
"propertyType": "CHOICE",
"requiredFlag": true
},
{
    "acceptableKeys": [
        {
            "displayName": "Alexa for Business",
            "keyName": "a4b",
            "scope": "SOURCE_AND_TARGET"
        },
        {
            "displayName": "Amazon API Gateway",
            "keyName": "apigateway",
            "scope": "SOURCE_AND_TARGET"
        },
        {
            "displayName": "Amazon AppStream 2.0",

```

```
        "keyName": "appstream2",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS AppSync Control Plane",
        "keyName": "appsync",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS AppSync Data Plane",
        "keyName": "appsync-api",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "Athena",
        "keyName": "athena",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Auto Scaling",
        "keyName": "autoscaling",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Batch",
        "keyName": "batch",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Cost Explorer",
        "keyName": "ce",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Cost and Usage Reports",
        "keyName": "cur",
        "scope": "SOURCE_AND_TARGET"
    },
    {
        "displayName": "AWS Budgets",
        "keyName": "budgets",
        "scope": "SOURCE_AND_TARGET"
    },
    {

```

```
    "displayName": "AWS Price List Service",
    "keyName": "api.pricing",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Certificate Manager",
    "keyName": "acm",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Certificate Manager Private Certificate Authority",
    "keyName": "acm-pca",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Cloud9",
    "keyName": "cloud9",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Cloud Directory",
    "keyName": "clouddirectory",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CloudFormation",
    "keyName": "cloudformation",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon CloudFront",
    "keyName": "cloudfront",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CloudHSM",
    "keyName": "cloudhsmv2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS CloudHSM Classic",
    "keyName": "cloudhsm",
    "scope": "SOURCE_AND_TARGET"
  },
  },
```



```
{
  "displayName": "Amazon CloudSearch",
  "keyName": "cloudsearch",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS CloudTrail",
  "keyName": "cloudtrail",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon CloudWatch",
  "keyName": "monitoring",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon CloudWatch Events",
  "keyName": "events",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Execute API",
  "keyName": "execute-api",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon CloudWatch Logs",
  "keyName": "logs",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS CodeBuild",
  "keyName": "codebuild",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS CodeCommit",
  "keyName": "codecommit",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS CodeDeploy",
  "keyName": "codedeploy",
  "scope": "SOURCE_AND_TARGET"
}
```

```
    },
    {
      "displayName": "AWS CodePipeline",
      "keyName": "codepipeline",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS CodeStar",
      "keyName": "codestar",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Cognito Your User Pools",
      "keyName": "cognito-idp",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Cognito Federated Identities",
      "keyName": "cognito-identity",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Cognito Sync",
      "keyName": "cognito-sync",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Comprehend",
      "keyName": "Comprehend",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Config",
      "keyName": "config",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Config Rules",
      "keyName": "configrules",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Connect",
      "keyName": "connect",
```

```
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Data Pipeline",
    "keyName": "datapipeline",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Database Migration Service",
    "keyName": "dms",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS DeepLens",
    "keyName": "deeplens",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Device Farm",
    "keyName": "devicefarm",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Direct Connect",
    "keyName": "directconnect",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Directory Service",
    "keyName": "ds",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon DynamoDB",
    "keyName": "dynamodb",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "DynamoDB Accelerator (DAX)",
    "keyName": "dax",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon DynamoDB Streams",
```

```
    "keyName": "streams.dynamodb",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elastic Beanstalk",
    "keyName": "elasticbeanstalk",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elastic Beanstalk Health Service",
    "keyName": "elasticbeanstalk-health",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic Compute Cloud (Amazon EC2)",
    "keyName": "ec2",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic Container Service",
    "keyName": "ecr",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic Container Service",
    "keyName": "ecs",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic Container Service for Kubernetes (Amazon EKS)",
    "keyName": "eks",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elastic File System",
    "keyName": "elasticfilesystem",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Elastic Load Balancing",
    "keyName": "elasticloadbalancing",
    "scope": "SOURCE_AND_TARGET"
  },
  {
```

```
    "displayName": "Amazon Elastic Transcoder",
    "keyName": "elastictranscoder",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon ElastiCache",
    "keyName": "elasticache",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Elasticsearch Service",
    "keyName": "es",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon EMR",
    "keyName": "elasticmapreduce",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Firewall Manager",
    "keyName": "fms",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon GameLift",
    "keyName": "gamelift",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Glacier",
    "keyName": "glacier",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Glue",
    "keyName": "glue",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Greengrass",
    "keyName": "greengrass",
    "scope": "SOURCE_AND_TARGET"
  },
  },
```

```
{
  "displayName": "Amazon GuardDuty",
  "keyName": "guardduty",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Health",
  "keyName": "health",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Identity and Access Management (IAM)",
  "keyName": "iam",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Import/Export",
  "keyName": "importexport",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Inspector",
  "keyName": "inspector",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS IoT 1-Click",
  "keyName": "iot1click",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS IoT Analytics",
  "keyName": "iotanalytics",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS IoT Core",
  "keyName": "iot",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Key Management Service",
  "keyName": "kms",
  "scope": "SOURCE_AND_TARGET"
}
```

```
    },
    {
      "displayName": "Amazon Kinesis Data Analytics",
      "keyName": "kinesisanalytics",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Kinesis Data Firehose",
      "keyName": "firehose",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Kinesis Data Streams",
      "keyName": "kinesis",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Kinesis Video Streams",
      "keyName": "kinesisvideo",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Lambda",
      "keyName": "lambda",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Lex (Model Building Endpoints)",
      "keyName": "models.lex",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Lex (Runtime Endpoints)",
      "keyName": "runtime.lex",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Lightsail",
      "keyName": "lightsail",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Macie",
      "keyName": "macie",
```

```
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Machine Learning",
    "keyName": "machinelearning",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaConvert",
    "keyName": "mediaconvert",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaLive",
    "keyName": "medialive",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaPackage",
    "keyName": "mediapackage",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaStore",
    "keyName": "mediastore",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Elemental MediaTailor",
    "keyName": "mediatailor",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Migration Hub",
    "keyName": "mgh",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Mobile Analytics",
    "keyName": "mobileanalytics",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon MQ",
```



```
    "keyName": "mq",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS OpsWorks (CM)",
    "keyName": "opsworks-cm",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS OpsWorks (Stacks)",
    "keyName": "opsworks",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Organizations",
    "keyName": "organizations",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Pinpoint",
    "keyName": "pinpoint",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Polly",
    "keyName": "polly",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon QuickSight",
    "keyName": "quicksight",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Redshift",
    "keyName": "redshift",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon Rekognition",
    "keyName": "rekognition",
    "scope": "SOURCE_AND_TARGET"
  },
  {
```

```
    "displayName": "Amazon Relational Database Service (Amazon RDS)",
    "keyName": "rds",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Resource Groups",
    "keyName": "resource-groups",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Secrets Manager",
    "keyName": "secretsmanager",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Server Migration Service",
    "keyName": "sms",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "Amazon SageMaker",
    "keyName": "api.sagemaker",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Security Token Service (AWS STS)",
    "keyName": "sts",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Serverless Application Repository",
    "keyName": "serverlessrepo",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Service Catalog",
    "keyName": "servicecatalog",
    "scope": "SOURCE_AND_TARGET"
  },
  {
    "displayName": "AWS Shield Advanced",
    "keyName": "Shield",
    "scope": "SOURCE_AND_TARGET"
  },
  },
```

```
{
  "displayName": "Amazon Simple Email Service (Amazon SES)",
  "keyName": "email-smtp",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Simple Notification Service (Amazon SNS)",
  "keyName": "sns",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Simple Queue Service (Amazon SQS)",
  "keyName": "sqs",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Simple Storage Service (Amazon S3)",
  "keyName": "s3",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Simple Storage Service Website Endpoints",
  "keyName": "s3-website",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon Simple Workflow Service (Amazon SWF)",
  "keyName": "swf",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "Amazon SimpleDB",
  "keyName": "sdb",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Snowball",
  "keyName": "snowball",
  "scope": "SOURCE_AND_TARGET"
},
{
  "displayName": "AWS Step Functions",
  "keyName": "states",
  "scope": "SOURCE_AND_TARGET"
}
```

```
    },
    {
      "displayName": "AWS Storage Gateway",
      "keyName": "storagegateway",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS Systems Manager",
      "keyName": "ssm",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Transcribe",
      "keyName": "transcribe",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Translate",
      "keyName": "Translate",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon Translate (FIPS)",
      "keyName": "translate-fips",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon VPC",
      "keyName": "waf",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon WorkDocs",
      "keyName": "workdocs",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "Amazon WorkSpaces",
      "keyName": "workspaces",
      "scope": "SOURCE_AND_TARGET"
    },
    {
      "displayName": "AWS X-Ray",
      "keyName": "xray",
```

```

        "scope": "SOURCE_AND_TARGET"
      }
    ],
    "displayName": "Service Name",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Service Name like S3",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "ServiceName",
    "propertyShortDesc": "Service Name",
    "propertyType": "CHOICE",
    "requiredFlag": true
  }
],
{
  "scope": "TARGET",
  "securityPolicy": "NONE",
  "securityProperties": []
},
{
  "scope": "SOURCE_AND_TARGET",
  "securityPolicy": "BASIC_AUTH",
  "securityProperties": [
    {
      "displayName": "Username",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "A username credential",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "username",
      "propertyShortDesc": "Use UserName",
      "propertyType": "STRING",
      "requiredFlag": true
    },
    {
      "displayName": "Password",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "A password credential",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "password",
      "propertyShortDesc": "Enter Password",
      "propertyType": "PASSWORD",

```

```

        "requiredFlag": true
    }
]
},
"securityProperties": [
    {
        "displayName": "Access Token URI",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.access.token.uri",
        "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
        "propertyType": "STRING",
        "propertyValue": "https://idcs-5ba32fa3496f48289532f8fc10f47032.identity.oraclecloud.com:443/oauth2/v1/token",
        "requiredFlag": true
    },
    {
        "displayName": "Client Id",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Used to identify the client(the software requesting an access token) that is making the
request. The value passed in this parameter must exactly match the value shown in your API console project.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.client.id",
        "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
        "propertyType": "STRING",
        "propertyValue": "<<your-IntegratedApp-ClientID>>",
        "requiredFlag": true
    },
    {
        "displayName": "Client Secret",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making the
request. The value passed in this parameter must exactly match the value shown in your API console project.",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "oauth.client.secret",
        "propertyShortDesc": "<unique random string matches your API console project>",
        "propertyType": "PASSWORD",
        "propertyValue": "*****",
        "requiredFlag": true
    }
],

```

```

{
  "displayName": "Scope",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "Permissions your application is requesting on behalf of the user.",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "oauth.scope",
  "propertyShortDesc": "for example: read,write.",
  "propertyType": "STRING",
  "propertyValue": "",
  "requiredFlag": false
},
{
  "displayName": "Auth Request Media Type",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "Auth Content-Type",
  "propertyGroup": "OTHER",
  "propertyName": "oauth.request.content.type",
  "propertyType": "STRING",
  "requiredFlag": false
},
{
  "displayName": "Username",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "A username credential",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "username",
  "propertyShortDesc": "Use UserName",
  "propertyType": "STRING",
  "propertyValue": "peter.obert+local@oracle.com",
  "requiredFlag": true
},
{
  "displayName": "Password",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "A password credential",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "password",
  "propertyShortDesc": "Enter Password",
  "propertyType": "PASSWORD",
  "propertyValue": "*****",

```

```
      "requiredFlag": true
    },
    {
      "acceptableKeys": [
        {
          "displayName": "Send client credentials as basic auth header",
          "keyName": "client_credentials_as_header",
          "scope": "SOURCE_AND_TARGET"
        },
        {
          "displayName": "Send client credentials in body",
          "keyName": "client_credentials_in_body",
          "scope": "SOURCE_AND_TARGET"
        }
      ],
      "displayName": "Client Authentication",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyDescription": "Client Authentication",
      "propertyGroup": "CREDENTIALS",
      "propertyName": "clientAuthentication",
      "propertyShortDesc": "Client Authentication",
      "propertyType": "CHOICE",
      "propertyValue": "client_credentials_as_header",
      "requiredFlag": false
    }
  ],
  "status": "CONFIGURED",
  "testStatus": false,
  "usage": 4,
  "usageActive": 1
}'
200 OK
```

Response

- Body
- Headers (12)

View More

```
json
{
  "adapterType": {
    "displayName": "REST",
```



```

    "links": [],
    "name": "rest",
    "type": "PREINSTALLED"
  },
  "agentRequired": false,
  "agentSupported": false,
  "connectionProperties": [
    {
      "acceptableKeys": [
        {
          "displayName": "REST API Base URL",
          "keyName": "restUrl",
          "scope": "SOURCE_AND_TARGET"
        },
        {
          "displayName": "Open API (1.0/2.0/3.0) URL",
          "keyName": "openApiUrl",
          "scope": "SOURCE_AND_TARGET"
        }
      ],
      "displayName": "Connection Type",
      "hasAttachment": false,
      "hiddenFlag": false,
      "propertyGroup": "CONNECTION_PROPS",
      "propertyName": "connectionType",
      "propertyType": "CHOICE",
      "propertyValue": "restUrl",
      "requiredFlag": true
    },
    {
      "acceptableKeys": [
        {
          "displayName": "TLSv1.1",
          "keyName": "TLSv1.1",
          "scope": "SOURCE_AND_TARGET"
        },
        {
          "displayName": "TLSv1.2",
          "keyName": "TLSv1.2",
          "scope": "SOURCE_AND_TARGET"
        }
      ],
      "displayName": "TLS Version",
      "hasAttachment": false,

```

```

        "hiddenFlag": false,
        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "tlsVersion",
        "propertyShortDesc": "TLS Version",
        "propertyType": "CHOICE",
        "requiredFlag": false
    },
    {
        "displayName": "Connection URL",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "connectionUrl",
        "propertyShortDesc": "Please make sure that this value really corresponds to the type selected above.",
        "propertyType": "URL",
        "propertyValue": "https://opa-oic-teamoic3-1740047240850-<<Storage-namespace>>-
fra.process.oci.oraclecloud.com/process/api",
        "requiredFlag": true
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Yes",
                "keyName": "true",
                "scope": "SOURCE_AND_TARGET"
            },
            {
                "displayName": "No",
                "keyName": "false",
                "scope": "SOURCE_AND_TARGET"
            }
        ],
        "displayName": "Enable two way SSL for outbound connections (Optional)",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Both the client and the server pass certificates to each other to establish an SSL link, when
two-way SSL is enabled. Both sides now know the identity of the other from their respective certificates.",
        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "enableTwoWaySSL",
        "propertyShortDesc": "Select \"Yes\" if both the client and server should authenticate each other.",
        "propertyType": "CHOICE",
        "requiredFlag": false
    },
    {

```

```

        "displayName": "Identity keystore alias name (Optional)",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Both the client and the server pass certificates to each other to establish an SSL link, when
two-way SSL is enabled. Name of the identity certificate alias to use for that communication. This value should match the alias
that was provided to import identity to ICS.",
        "propertyGroup": "CONNECTION_PROPS",
        "propertyName": "sslCertificateAlias",
        "propertyShortDesc": "Name of alias to use for establishing identity during two way SSL communication",
        "propertyType": "STRING",
        "requiredFlag": false
    }
],
"created": "2025-03-07T15:08:51.108+0000",
"description": "Invoking the process automation to create new process instances",
"hasOverride": false,
"id": "POB_OICTOOPA_INVOKE",
"keywords": "",
"lastUpdated": "2025-03-07T15:08:51.119+0000",
"lastUpdatedBy": "<<your-IntegratedApp-ClientID>>",
"links": [
    {
        "href": "https://design.integration.<<your-OIC-Instance-
region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/POB_OICTOOPA_INVOKE?integrationInstance=teamoic3-<<Storage-
namespace>>-fr",
        "rel": "self"
    },
    {
        "href": "https://design.integration.<<your-OIC-Instance-
region>>.ocp.oraclecloud.com/ic/api/integration/v1/connections/POB_OICTOOPA_INVOKE?integrationInstance=teamoic3-<<Storage-
namespace>>-fr",
        "rel": "canonical"
    }
],
"lockedFlag": false,
"name": "pob-OICtoOPA-Invoke",
"percentageComplete": 100,
"privateEndpoint": false,
"privateEndpointSupported": false,
"role": "TARGET",
"securityPolicy": "OAUTH_RESOURCE_OWNER_PASSWORD_CREDENTIALS",
"securityPolicyInfo": [],
"securityProperties": [
    {

```

```

    "displayName": "Access Token URI",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "A request should be sent to this URI for obtaining an access token.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.access.token.uri",
    "propertyShortDesc": "for example: https://accounts.google.com/o/oauth2/token",
    "propertyType": "STRING",
    "propertyValue": "https://idcs-5ba32fa3496f48289532f8fc10f47032.identity.oraclecloud.com:443/oauth2/v1/token",
    "requiredFlag": true
  },
  {
    "displayName": "Client Id",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Used to identify the client(the software requesting an access token) that is making the
request. The value passed in this parameter must exactly match the value shown in your API console project.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.client.id",
    "propertyShortDesc": "for example: 123-xxx.apps.googleusercontent.com",
    "propertyType": "STRING",
    "propertyValue": "<<your-IntegratedApp-ClientID>>",
    "requiredFlag": true
  },
  {
    "displayName": "Client Secret",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Used to authorize the client(the software requesting an access token) that is making the
request. The value passed in this parameter must exactly match the value shown in your API console project.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.client.secret",
    "propertyShortDesc": "<unique random string matches your API console project>",
    "propertyType": "PASSWORD",
    "requiredFlag": true
  },
  {
    "displayName": "Scope",
    "hasAttachment": false,
    "hiddenFlag": false,
    "propertyDescription": "Permissions your application is requesting on behalf of the user.",
    "propertyGroup": "CREDENTIALS",
    "propertyName": "oauth.scope",
    "propertyShortDesc": "for example: read,write.",
  }

```

```

        "propertyType": "STRING",
        "propertyValue": "https://opa-oic-teamoic3-1740047240850-<<Storage-namespace>>-fra.process.oc1.oraclecloud.com/process",
        "requiredFlag": false
    },
    {
        "displayName": "Auth Request Media Type",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "Auth Content-Type",
        "propertyGroup": "OTHER",
        "propertyName": "oauth.request.content.type",
        "propertyType": "STRING",
        "requiredFlag": false
    },
    {
        "displayName": "Username",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "A username credential",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "username",
        "propertyShortDesc": "Use UserName",
        "propertyType": "STRING",
        "propertyValue": "peter.obert+local@oracle.com",
        "requiredFlag": true
    },
    {
        "displayName": "Password",
        "hasAttachment": false,
        "hiddenFlag": false,
        "propertyDescription": "A password credential",
        "propertyGroup": "CREDENTIALS",
        "propertyName": "password",
        "propertyShortDesc": "Enter Password",
        "propertyType": "PASSWORD",
        "requiredFlag": true
    },
    {
        "acceptableKeys": [
            {
                "displayName": "Send client credentials as basic auth header",
                "keyName": "client_credentials_as_header",
                "scope": "SOURCE_AND_TARGET"
            }
        ]
    }

```

```

    },
    {
      "displayName": "Send client credentials in body",
      "keyName": "client_credentials_in_body",
      "scope": "SOURCE_AND_TARGET"
    }
  ],
  "displayName": "Client Authentication",
  "hasAttachment": false,
  "hiddenFlag": false,
  "propertyDescription": "Client Authentication",
  "propertyGroup": "CREDENTIALS",
  "propertyName": "clientAuthentication",
  "propertyShortDesc": "Client Authentication",
  "propertyType": "CHOICE",
  "propertyValue": "client_credentials_as_header",
  "requiredFlag": false
},
"status": "CONFIGURED",
"testStatus": false,
"usage": 4,
"usageActive": 1
}

```

Integrations

This folder contains the most frequently used [API requests related to integration flows](#) management.

Typical commands used to deploy integration flows between the environments are to get(export) integration details, export integration, import integration, activate and deactivate and configure integrationcon.

AuthorizationOAuth 2.0

This folder is using an authorization helper from collection CICD-OIC3-quickstart

DeactivateConfigureActivate

This folder contains requests that could be used to configure(re-link) connections if imported connection works with different connections in different environment.

AuthorizationOAuth 2.0

This folder is using an authorization helper from collection CICD-OIC3-quickstart

POSTOIC3_DEV_Integration_DeActBeforeReConfigure

[Open request](#)

https://design.integration.<<your-OIC-Instance-region>>.ocp.oraclecloud.com/ic/api/integration/v1/integrations/<<your-IntegrationID>>|01.00.0001?integrationInstance=<<your-OIC-InstanceName>>

Add request description...

AuthorizationOAuth 2.0

Query Params

integrationInstance

<<your-OIC-InstanceName>>

enableAsyncActivationMode

true

Bodyraw (json)

```
json
{
  "status": "CONFIGURED"
}
```

Example

OIC3_DEV_Integration_DeActivate_<<your-integration-name>>

Request

[View More](#)

cURL

```
curl --location 'https://design.integration.<<your-OIC-Instance-region>>.ocp.oraclecloud.com/ic/api/integration/v1/integrations/<<your-integration-name>>|01.00.0000?integrationInstance=%3C%3Cyour-OIC-InstanceName%3E%3E&enableAsyncActivationMode=true' \
--header 'X-HTTP-Method-Override: PATCH' \
--data '{
  "status": "CONFIGURED"
}'
200 OK
```

Response

- Body
- Headers (11)

```
json
{
  "apiDeploymentInProgress": false,
  "compatible": false,
  "links": [],
  "lockedFlag": false,
  "scheduleApplicableFlag": false,
  "scheduleDefinedFlag": false,
  "status": "DEACTIVATION_INPROGRESS",
  "tempCopyExists": false,
  "totalEndPoints": 0
}
```

POSTOIC3_DEV_Integration_Configure

Open request

```
https://design.integration.<<your-OIC-Instance-region>>.ocp.oraclecloud.com/ic/api/integration/v1/integrations/<<your-IntegrationID>>|01.00.0001?integrationInstance=<<your-OIC-InstanceName>>
```

Add request description...

AuthorizationOAuth 2.0

Query Params

integrationInstance

<<your-OIC-InstanceName>>

Bodyraw (json)

View More

```
json
{"id": "<<your-IntegrationID>>|01.00.0001", "dependencies": {"connections": [{"id": "REST_TRIGGER"}, {"id": "<<your-IntegrationID>>"}], "certificates": []}}
```

Example

OIC3_DEV_Integration_Activate_<<your-integration-name>> Copy

Request

View More

cURL

```
curl --location 'https://design.integration.<<your-OIC-Instance-region>>.ocp.oraclecloud.com/ic/api/integration/v1/integrations/<<your-integration-name>>|01.00.0000?integrationInstance=%3C%3Cyour-OIC-InstanceName%3E%3E' \
--header 'X-HTTP-Method-Override: PATCH' \
--data '{"id": "<<your-IntegrationID>>|01.00.0001", "dependencies": {"connections": [{"id": "REST_TRIGGER"}, {"id": "<<your-IntegrationID>>"}], "certificates": []}}'
412 PRECONDITION FAILED
```

Response

- Body
- Headers (11)

View More

```
json
{
  "status": "HTTP 412 Precondition Failed",
  "title": "Failed to save integration configuration because it is active or is in the process of being activated.",
  "type": "https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.4.13"
}
```

POSTOIC3_DEV_Integration_ActAfterConfig

Open request

https://design.integration.<<your-OIC-Instance-region>>.ocp.oraclecloud.com/ic/api/integration/v1/integrations/<<your-IntegrationID>>|01.00.0001?integrationInstance=<<your-OIC-InstanceName>>&enableAsyncActivationMode=true

Add request description...

Authorization

OAuth 2.0

Query Params

integrationInstance
<<your-OIC-InstanceName>>
enableAsyncActivationMode
true

Body

raw (json)

```
json
{
  "status": "ACTIVATED"
}
```

Example

OIC3_DEV_Integration_DeActivate_<<your-integration-name>>

Request

View More

cURL

```
curl --location 'https://design.integration.<<your-OIC-Instance-
region>>.ocp.oraclecloud.com/ic/api/integration/v1/integrations/<<your-integration-
name>>|01.00.0000?integrationInstance=%3C%3Cyour-OIC-InstanceName%3E%3E&enableAsyncActivationMode=true' \
--header 'X-HTTP-Method-Override: PATCH' \
--data '{
  "status": "CONFIGURED"
}'
200 OK
```

Response

- Body
- Headers (11)

```
json
{
  "apiDeploymentInProgress": false,
  "compatible": false,
  "links": [],
  "lockedFlag": false,
  "scheduleApplicableFlag": false,
```

```
"scheduleDefinedFlag": false,  
"status": "DEACTIVATION_INPROGRESS",  
"tempCopyExists": false,  
"totalEndpoints": 0  
}
```

Scheduled_Integrations

This folder contains requests that could be base for creating, starting, stopping and deleting [schedules for the scheduled integration flows](#) which can be in project or global.

AuthorizationOAuth 2.0

This folder is using an authorization helper from collection CICD-OIC3-quickstart

JUMP TO

Introduction

AccessToken

Connections

Integrations

Lookups

Projects

OIC Instance Export/Import