Identity Registration Web Service REST API v2

The IRWS v2 documentation is not exhaustive. Please contact iam-support@uw.edu if you have questions about v2 usage.

Overview

This guide describes the Identity Registration Web Service (IRWS) API V2. IRWS v2 is deprecated and all new customers should use IRWS v3 instead . I RWS can be used to to register an identity with the Identity Registry. IRWS offers a "RESTful" programmatic interface. It exposes Identity Registry source information as addressable resources via the uniform HTTP interface; authorized clients may retrieve (GET), create (PUT), update (POST) and delete (DELETE) representations of these resources through the REST API.

Request IRWS Access

IRWS v2 Swagger Documentation

Swagger documentation for IRWS v2 can be found at: Identity Registration Web Service REST API v2.

Requirements

The application that uses this service must support:

- HTTPS
- Authentication with X.509 certificates
- HTTP GET, PUT, POST, and DELETE requests
- JSON content type only

Base URL

All interactions with IRWS are through the following base URLs. Customer testing is generally done in the Eval environment. The Dev environment is reserved for the IAM development team.

Environment	URL
Development	https://mango-dev.u.washington.edu:646/registry/v2
Evaluation	https://mango-eval.u.washington.edu:646/registry/v2
Production	https://mango.u.washington.edu:646/registry/v2

Supported HTTP Request Headers

IRWS supports a set of standard and custom HTTP headers that can be used by a client to express options. All the listed request headers are optional and all can be combined.

```
Accept: [content type1,content type2, ...]
```

- This is a standard header that works as expected for IRWS.
- IRWS V2 accepts payloads and returns content only in JSON format.
- Specifying an unsupported content type will return a 406 return code.
- If your ACCEPT header doesn't include any content types, IRWS will use the default content type for the resource.
- Example: `Accept: application/json`

`Act-As: application name`

- Act-As allows authorized clients to act as another client.
- Contact us at iam-support@uw.edu for help with configuring Act-As.

`Application-Name: application/userid`

- Application-Name specifies the label associated with a clients requests in the logs.
- Contact us at iam-support@uw.edu for help with selecting your application name.

`Attribute-List: [attribute1 attribute2 attribute3 ...]`

- Doing a GET on a resource will generally return all attributes in that resource, which can be a long list. If a client is only interested in a small subset of attributes, this header can be used to limit the return payload to those attributes.
- Example: Attribute-List: lname fname wp_email

`If-Match: [etag]`

- This is a standard header that works as expected for IRWS.
- Example: `Etag: 20150425080640Z`

`Option-List: [imprecise pretend pretty reflect]`

- `imprecise` - Overrides default PUT behavior to behave like a POST if the record already exists. This is the default behavior for some resources.
- `pretend` - Reports what would happen if the request was carried out, but the changes are not committed. Useful for testing.
- `pretty` - By default, IRWS returns a payload as a single line of text. For human readability you can specify this option to get a pretty-printed representation of the data.
- `reflect` - Useful for PUTs and POSTS, specifying reflect will cause IRWS to return the current state of the resource. This is the default behavior for the `source` and `social` resources.
- Example: `Option-List: imprecise reflect`

Supported URI Parameters

HTTP headers are the preferred way for an application to express the IRWS options it needs. For convenience, some of the options described above can be communicated via parameters in the URI. This may be useful for user interactive testing with a browser or the curl utility. All parameters are optional and different options may be combined in one URI. Definitions are provided in the previous section.

- `-type=[json]`
- `-actas=permit=[application name]`
- `-attribute=[attribute1+attribute2...]`
- `-imprecise`
- `-pretend`
- `-pretty`
- `-reflect`
- Examples:
    - `GET https://mango-eval.u.washington.edu:646/registry-eval/v1/person/uwhr/123456789&-type=json&-attribute=lname+fname+wp_email&-pretty`

Resources

Identity records are associated with a resource. A GET on the base URL will return a list of all top-level resources for which your client is authorized. Documentation is provided for the resources listed in the table below.

Resource	Description
Account	The Account resource allows an authorized client to create, read, update, delete, and perform other actions related to account entities.
Category	The Category resource allows an authorized client to...
Dsubscription	The Dsubscription resource allows an authorized client to...
Name	The Name resource allows an authorized client to read and update source-independent names associated with person entities.
Pdsentry	The PDS Entry resource allows an authorized client to read Person Directory data.
Person	The Person resource allows an authorized client to create, read, update, delete, and perform other actions related to person entities.
Regid	The Regid resource allows an authorized client to manipulate attributes associated with regids.
Social	The Social resource allows an authorized client to create, read, and update social identities.
Subscription	The Subscription resource allows an authorized client to...

Return Codes

IRWS uses standard HTTP return codes, as well as a few special ones. Most errors will return a JSON payload with the error class containing the `code` and `message` properties.

Return Code	Description	Payload
200	OK (the GET, PUT, or POST was successful)	Resource (JSON)
400	Invalid identifier type provided	Error (JSON)
401		
403		
404	Identifier provided did not match any records	Error (JSON)
405	Method Not Allowed	Error (JSON)
406	Content type not supported	
409		
412		
500	Internal Server Error	None
501	Not Implemented	None
503	Service Unavailable	None
901		
902		
903		
904		