

ID Match PoC

- [Gaining Access](#)
- [Data](#)
- [Attributes](#)
- [Matching Rules](#)
 - [Canonical Rules](#)
 - [Potential Rules](#)
- [Sending Queries](#)
 - [Search/Update](#)
 - [Search Only](#)
 - [Notes](#)
- [Performance Notes](#)

IdMatch is a method of determining if a person presented by a System of Record is already known to the Identity Management System.

A demonstration version of the PostgreSQL-based [ID match engine](#) is available to anyone who wishes to experiment with it. This engine implements the [CI FER ID Match Strawman API](#).

Gaining Access

The API endpoint is <https://idmatch.testbed.tier.internet2.edu/match-poc>

Access is via Basic Auth over https. Credentials will be assigned to a "System of Record" for use in making requests.

Data

The match engine is preloaded with 150,000 fake records, which you can download [here](#) if you wish to craft queries against them. These records are all loaded via the "test" System of Record.

Attributes

The match engine (as configured for this demo) supports the following attributes (along with their API labels):

- System of Record (SOR; used to construct API resource path, assigned with credentials)
- SOR ID (used to construct API resource path and to unique identify the subject within the SOR)
- SSN (`identifiers/identifier` of type `national`, use 000-00-0000 or other appropriate national ID format)
 - Punctuation is ignored (so 000-00-0000 and 000000000 are the same)
 - 000-00-0000 is considered null (and ignored)
 - For potential matches, a distance of 2 is permitted (meaning 000-12-0000 would be a candidate match for 000-21-0000)
- Given Name (`names/given` of type `official`)
 - For potential matches, a distance of 2 is permitted
 - For potential matches, a substring match of the first three characters is permitted
- Family Name (`names/family` of type `official`)
 - For potential matches, a distance of 2 is permitted
- Date of Birth (`dateOfBirth`, use YYYYMMDD format)
 - Stored as a string (so 2016-09-26 and 2016-9-26 are *not* the same)
 - Punctuation is ignored (so 2016-09-26 and 20160926 are the same)
 - 0000-00-00 is considered null (and ignored)
 - For potential matches, a distance of 2 is permitted
- Email Address (`emailAddresses/mail` of type `personal`)

Matching Rules

The following rules are configured for this demo:

Canonical Rules

If any of these rules finds exactly one match, processing stops and that match is returned. If any rule finds more than one match, the response is automatically switched to a potential match.

Canonical rules must match each attribute exactly.

1. SOR + SORID
2. SSN + Last Name + DoB

3. SSN + Last Name + First Name

Potential Rules

1. SSN + Last Name + DoB(distance)
2. SSN(distance) + Last Name + DoB
3. SSN + Last Name + First Name(distance)
4. SSN(distance) + Last Name + First Name
5. DoB + Last Name(distance) + First Name(distance)
6. DoB + Last Name(distance) + Substring(distance)
7. Email

Sending Queries



DO NOT SEND REAL DATA TO THIS SERVICE.

Send fake data only.

The full specification for sending queries is available in the [CIFER ID Match Strawman API](#), however you will most likely want to send one of two requests.

Search/Update

Corresponds to the [Reference Identifier Request](#).

- Replace SOR with the SOR label you were assigned.
- Replace SORID with your internal identifier for this record.

Returns:

- 200: Existing record found
- 201: New record created
- 300: Multiple candidates

```
PUT https://idmatch.testbed.tier.internet2.edu/match-poc/v1/people/SOR/SORID
{
  "sorAttributes":
  {
    "names": [
      {
        "given": "Pat",
        "family": "Lee",
        "type": "official"
      }
    ],
    "dateOfBirth": "1983-03-18",
    "identifiers": [
      {
        "type": "national",
        "identifier": "012-99-5678"
      }
    ]
  }
}
```

Search Only

Corresponds to the [Reference Identifier Request \(Search Only\)](#). Parameters are as for the Search/Update request, above.

Returns:

- 200: Existing record found
- 300: Multiple candidates
- 404: No match found

```
POST https://idmatch.testbed.tier.internet2.edu/match-poc/v1/people/SOR/SORID
{
  ... as above ...
}
```

Notes

1. If you successfully submit a Reference Identifier (PUT) Request (that is, you receive a reference identifier back in response), and you then try to issue the exact same request you will not receive the reference identifier back in subsequent responses. This is because your subsequent requests are technically [attribute update requests](#).

Performance Notes

The demonstration server is a stock testbed VM running CentOS 7 and a local version of PostgreSQL 9. No optimizations have been made to the hardware, virtual machine, or database configuration other than the creation of indexes as describe the installation instructions.

This table shows the processing time for the initial data load. Note that in order for a new record to be added, all defined match rules must execute (and return no matches) – that is the worst possible performance for a query.

Load #	Records	Total Time (sec)	Average Time Per Record (ms)	Non-Exact Results
1	10000	561	56.1	0
2	40000	2601	65.0	3
3	50000	6079	121.6	335
4	50000	11874	237.5 (max ~1187)	532

Performance numbers reflect queries performed against localhost. Remote queries will take longer due to network latency.

The increasing average request time is likely due to the increased expense of updating the database indexes as the data set gets larger.