

Overview

There are scenarios where customers using SheerID-hosted verify applications may want to pass data into the verification form to reduce duplicate data entry. This can be accomplished by submitting either a GET or POST HTTP request using the verification template URL. Please note that there is currently no protection against the user changing data that has been pre-populated. This technique should be understood as merely a convenience to the user and not a mechanism for conveying tamper-proof data.

Conventions

Template URL

The template URL used in the examples below is the simple form (based on the template ID). The GET and POST techniques work equally well with friendly (namespace-based) URLs as well. This is the same URL that you see in your browser when accessing the verification page.

Parameter Names

The data passed via query-string or POST data must use the following parameter names:

- type - the affiliation type, should match up with one of the available radio button options
- data[$\$(name)$] - Person fields are supplied as a data object, where $\$(name)$ is replaced with the field name. (See [List Fields](#) documentation for a list of available fields)
- metadata[$\$(name)$] - Request metadata are supplied as a metadata object, where $\$(name)$ is replaced with the meta key. In order to supply metadata, the template must be configured with `extraFields` directive that matches the keys supplied here

Any of these fields are optional, if they are supplied the form will be pre-populated with the supplied value.

Usage

Example Template

For demonstration purposes, we'll start by creating the template that will receive pre-populated data. This can be done via the Control Center application or via REST API as shown below. In many cases, SheerID may have provisioned this for you.

```
$ curl -H "Authorization: Bearer $TOKEN" https://services-sandbox.sheerid.com/rest/0.5/template \
-d name=StudentTest -d _affiliationTypes=STUDENT_FULL_TIME,STUDENT_PART_TIME \
-d organizationType=UNIVERSITY -d extraFields=color
```

```
{
  "active": true,
  "config": {
    "affiliationTypes": [
      "STUDENT_FULL_TIME",
      "STUDENT_PART_TIME"
    ],
    "assetTypes": [
      "TUITION_RECEIPT",
      "TRANSCRIPT",
      "CLASS_SCHEDULE",
      "OFFICIAL_LETTER",
      "OTHER",
      "DATED_ID_CARD",
      "REGISTRATION_RECEIPT",
      "ID_CARD"
    ],
    "consolationRewardIds": [],
    "metadata": {},
    "notifierIds": null,
    "rewardIds": [],
    "verificationTypes": [
      "AUTHORITATIVE",
      "ASSET_REVIEW"
    ]
  },
  "expires": null,
  "id": "54f9b99be4b05abe94f78a8c",
  "metadata": {
    "extraFields": "color",
    "organizationType": "UNIVERSITY"
  },
  "name": "StudentTest"
}
```

POST

Parameters are passed to the verify page via POST data. In both GET and POST strategies, all information is transmitted securely using HTTPS. The benefit of this strategy is that any supplied information is not present in the URL, which prevents the exposure of the supplied data to server logs or end-user bookmarks.

```
<form action="https://services-sandbox.sheerid.com/verify/54f9b99be4b05abe94f78a8c/" method="POST">
  <input type="hidden" name="action" value="retry" />
  <input type="hidden" name="type" value="STUDENT_FULL_TIME" />
  <input type="hidden" name="data[FIRST_NAME]" value="Test" />
```

```
<input type="hidden" name="data[LAST_NAME]" value="User" />
<input type="hidden" name="data[EMAIL]" value="test@sheerid.com" />
<input type="hidden" name="metadata[color]" value="blue" />
<button type="submit">Go</button>
</form>
```

GET

Parameters may be passed to the verify page using a query string. The advantage of this strategy is its simplicity.

[https://services-sandbox.sheerid.com/verify/54f9b99be4b05abe94f78a8c/?type=STUDENT_FULL_TIME&data\[FIRST_NAME\]=Test&data\[LAST_NAME\]=User&data\[EMAIL\]=test@sheerid.com](https://services-sandbox.sheerid.com/verify/54f9b99be4b05abe94f78a8c/?type=STUDENT_FULL_TIME&data[FIRST_NAME]=Test&data[LAST_NAME]=User&data[EMAIL]=test@sheerid.com)