

Overview

SheerID's Asset Review verification source allows end-users to supply photographs/scans of documentation proving their affiliation, such as a membership card, transcript, pay stub etc. The direct asset upload feature of the API allows end-users to submit their documents directly to SheerID to ensure they are transferred quickly and securely into SheerID's review system, without the integrator having to act as an intermediary for the file transfer.

Techniques

Below is a list of techniques that can be employed to take advantage of the direct asset upload. Note that in each of these cases, you will need to have performed the following steps:

1. Perform a verification using the [Verify](#) method
2. Create an asset upload token using the [Issue Token](#) method

Also note that the URLs shown in this document refer to the production environment. The same techniques can be used in the sandbox environment by replacing the hostname `services.sheerid.com` with `services-sandbox.sheerid.com` where appropriate.

Hosted Upload Page

Link to the SheerID-hosted Upload page, including your asset upload token in the query string as shown:

```
https://services.sheerid.com/upload/?token=${assetUploadToken}
```

Custom Form

Configure a form on your site that posts directly to the SheerID API. Ensure that your form is configured with an `enctype` attribute set to `multipart/form-data`.

```
<form enctype="multipart/form-data" method="POST" action="https://services.sheerid.com/rest/0.5/asset">
  <input type="hidden" name="token" value="${assetUploadToken}"/>
  <input name="success" type="hidden" value="${successUrl}"/>
  <input name="failure" type="hidden" value="${failureUrl}"/>
  <label>File: </label><input type="file" name="file"/>
  <button type="submit">Submit</button>
</form>
```

Form elements:

- `token` - your asset upload token
- `file` - a file-upload field that represents the file data. Note: you may upload multiple files at once by supplying additional parameters prefixed by `file` (example: `file0`, `file1`, etc.)
- `successUrl` - a URL to which the user is redirected upon successful upload.
- `failureUrl` - a URL to which the user is redirected upon upload failure or configuration error. A

query string parameter, `error`, will be appended to this URL to provide more information about the nature of the error.

- `?error=400` - required fields are missing or invalid, file field is empty
- `?error=401` - the token is invalid or expired
- `?error=500` - unexpected server error

Javascript API

SheerID provides a Javascript snippet which renders a form similar to that shown in the previous example.

```
<div id="upload"></div>
<script src="https://www.sheerid.com/jsapi/js/SheerID.js"></script>
<script>
SheerID.load('asset', '1.0', {
  config: {
    container: 'upload', // the ID of an element in the page inside of which the form should be rendered
    maxFiles: 3, // the maximum number of files that can be uploaded at a time
    success : '${successUrl}', // replace with your success URL
    failure : '${failureUrl}', // replace with your failure URL
    ajax: true, // whether to use "AJAX"-style upload, see discussion below
    token: '${assetUploadToken}' // replace with your asset upload token
  }
});
</script>
```

A note about AJAX mode

AJAX-style uploads can be used to keep the end-user on the same page while the upload is processing and after it has completed. The background file upload is not actually using AJAX but rather POSTing to a hidden iframe and waiting for the result. The success and failure URLs in this case are not displayed to the user, but instead used as indicators of the success/failure status of the upload request. For this reason, it is very important that these URLs match completely the domain, port and protocol of the page serving the upload form, in order for the event handlers to properly detect a completed upload. With AJAX mode disabled (this is the default configuration), the form will send the user to the redirect pages as described in the previous example.