SheerID HTTP Notifier Signature

Overview

SheerID HTTP notifiers (webhooks) send HTTPS requests to a pre-configured URL on your domain to notify your system of various SheerID verification events. In order to ensure authenticity and integrity of the notification, we attach a signature as an HTTP header to each of these requests.

Notification Data

`POST` With Form Data Format

A HTTP notifier by default or configured with `postDataType=formData` uses the Form Data format with `application/x-www-form-urlencoded` data. HTTP `POST` notifications using Form Data format will contain the following data in the `POST` body:

```
requestId=${requestId}
```

- `requestId` - an identifier for the verification request for which this event was triggered

A HTTP notifier configured with `extraSigningFields=true` will also contain `timestamp` and `nonce` in the `POST` body:

```
requestId=${requestId}&timestamp=${utc_timestamp_in_milliseconds}&nonce=${nonce}
```

- `timestamp` - a Unix timestamp (UTC time), with millisecond resolution
- `nonce` - a randomly-generated, single use token used to add additional entropy to the raw POST data

`POST` With JSON Format

A HTTP notifier configured with `postDataType=json` uses the JSON format with `application/json` data. HTTP `POST` notifications using Form Data format will contain the following data in the `POST` body:

```
{ "requestId" : "${requestId}" }
```

- `requestId` - an identifier for the verification request for which this event was triggered

A HTTP notifier configured with `extraSigningFields=true` will also contain `timestamp` and `nonce` in the `POST` body:

```
{ "requestId" : "${requestId}", "timestamp" : ${utc_timestamp_in_milliseconds}, "nonce"
: "${nonce}" }
```

- `timestamp` - a Unix timestamp (UTC time), with millisecond resolution
- `nonce` - a randomly-generated, single use token used to add additional entropy to the raw POST

data

GET

Signatures are not supported for HTTP `GET` notifications. We recommend that the HTTP `POST` method is used instead.

Implementation

All HTTP `POST` notifications originating from SheerID contain a header, `X-SheerID-Signature` whose value is a HMAC-SHA256 digest of the notification data, according to the steps below.

1. Obtain your account's Secret Token (can be found on the API Access Tokens page in SheerID Control Center and is **ONLY** accessible by the account owner)
2. Obtain the notification data exactly as provided in the `POST` body of the request (UTF-8 encoded, `application/x-www-form-urlencoded` or `application/json` data)
3. Create a HMAC-256 digest of the notification data (UTF-8 encoded), with the (UTF-8 encoded) Secret Token used as the secret key
4. The digest obtained in step 3 above is attached as a request header (`X-SheerID-Signature`)NOTE: If the generated HMAC-256 digest does not match the disgest obtained from the (`X-SheerID-Signature`) request header, please verify the notification data matches the exact format as the data from the `POST` body. Any variation between the notification data and the data from the `POST` body, such as adding or removing a space, will result in an unexpected digest value.

Signature Validation

Recipients of HTTP notifier notifications from SheerID **SHOULD** verify the signature of each notification received to ensure its authenticity and integrity. This can be accomplished by following the Implementation steps noted above based on the message received, and asserting that the resulting digest obtained matches the value of the `X-SheerID-Signature` header.

Code sample (python)

~~~
````
import hashlib
import hmac

digest = hmac.new(client_secret.encode('utf-8'), msg=raw_response.encode('utf-8'),
digestmod=hashlib.sha256).hexdigest()
````
~~~