



EchoSign API Guide

This document provides an overview of how to use the EchoSign API in common integration scenarios.

Table of Contents

Sending a document for Signature.....	2
Checking the status of a document	2
Retrieving the final signed document.....	3
Signing Widgets.....	4
Checking the status of a document signed through a widget.....	4



[Sending a document for Signature](#)

To send a document out for signature through the EchoSign API, call the `sendDocument` method. This method has 3 parameters that you will need to pass, `apiKey`, `senderInfo` and `documentCreationInfo`. For more information on EchoSign API methods, please refer to the [API documentation](#).

After you send a document, you can retrieve up-to-date status of the document either by polling, or when EchoSign notifies you the status of the document has changed.

Checking the status of a document

There are two ways you can retrieve up-to-date status of the document: by polling, or by calling the `getDocumentInfo` method when EchoSign notifies you the status of the document has changed.

1. Polling

You can call the `getDocumentInfo` method periodically to check the status of your documents.

The upside of polling is that it is more reliable in the event that callback fails since your system is down, scheduled maintenance, lack of connectivity, etc.

The downside is that you have to create a scheduling mechanism to query the status of all documents that were not yet signed, waste time and resources checking to see if a document status has changed instead of only retrieving the updated status when you know it has changed.

If you choose to use polling, we recommend you will have different policies based on document "age" - ie. reducing the frequency of polling if the document was not signed after X days.

2. Ping Notifications

Ping notifications allow your application to retrieve up-to-date status after EchoSign notifies you the status of a document had changed.

In the `CallbackInfo` object used in `DocumentCreationInfo`, you can set the `signedDocumentUrl` parameter. This is the URL to which EchoSign will call your system using HTTP GET every time the status of the document has changed. You can then use `getDocumentInfo` to get the latest status of the



document.

You probably have a unique identifier on your end for each document, and some sort of table that maps that to the documentKey you get as an output from sendDocument. Pass your unique identifier as a parameter in the URL, so that when EchoSign pings you to notify the document status has changed, you will know which document ID on your end we are talking about, and subsequently which documentKey.

Then, call the getDocumentInfo method and provide that documentKey, and EchoSign will return the updated status of the document.

In most cases, when the document status is "Signed", you would like to retrieve the signed document PDF using the getLatestDocument method and store it in your system.

The HTTP GET method of retrieving up-to-date document status is not configured by default. Please contact EchoSign if you wish to use this option.

Retrieving the final signed document

There are two ways you can retrieve the final signed document: calling getLatestDocument after the document has been signed, or by setting the signedDocumentUrl where EchoSign will do a HTTP PUT with the final signed PDF if you aren't using the callback mechanism to notify you when document status has changed.

1. Call getLatestDocument after the document has been signed

If you've determined the document has been signed, by calling the getDocumentInfo method for example, you can then call the getLatestDocument method to retrieve the final signed PDF.

2. Setting the signedDocumentUrl where EchoSign will do a HTTP PUT with the final signed PDF

In the CallbackInfo object used in DocumentCreationInfo, you can set the signedDocumentUrl parameter. This is the URL to which EchoSign will do a HTTP PUT with the final signed PDF.

Note: HTTP PUT is the default behavior on the signedDocumentUrl, but if you configure HTTP GET to check the status of a document, this option for receiving the final signed document is no longer



available.

Signing Widgets

Checking the status of a document signed through a widget

A widget is a template. Every document signed through a widget is a separate instance of that widget, and EchoSign creates a unique documentKey for every such document. A good way to think about the relationship of the widget and the documents signed through it is a parent-child relationship. Therefore, the status of the “parent” widget itself does not reflect the status of its children.

Using this terminology, the documentKey of the widget is the parent documentKey. The documentKeys for every document signed through the widget are the child documentKeys. The child documentKeys are what you need to use when checking the status of a specific document signed through a widget.

There are two ways you can find the child documentKeys: by using the getFormData method, or by using the completion URL.

1. Finding the child documentKey using the getFormData method

Call the getFormData method and pass it the documentKey of the parent widget. The output will include the data in comma-separated value (CSV) format. The first line include column header names and then a line for each signer who signed the parent widget.

The document keys of all child widgets will be in the first column, under “EchoSign Transaction number”. See example below:

```
EchoSign transaction number, Agreement name, signed, email  
12ABC3D456E7F,test widget,2/5/10 09:21,eran@echosign.com
```



Notice there may be multiple child agreements to the same parent Widget - each with a different document key. Each child document can be signed by one or two signers (the original signer and potentially by the sender), resulting in multiple rows in the CSV. See example below:

```
EchoSign transaction number, Agreement name, signed, email  
12ABC3D456E7F,test widget,2/5/10 09:21,eran@echosign.com  
98ZYX7W654V3U,test widget,2/6/10 11:56,eran2@echosign.com
```

If the child document is signed by two signers, there will be two rows in the CSV with the same document key. See example below:

```
EchoSign transaction number, Agreement name, signed, email  
12ABC3D456E7F,test widget,2/5/10 09:21,eran@echosign.com  
98ZYX7W654V3U,test widget,2/6/10 11:56,eran2@echosign.com  
12ABC3D456E7F,test widget,2/6/10 13:37,eran3@echosign.com
```

2. Finding the child documentKeys using the completion URL

If the widget is part of a multi-step process, you can tell EchoSign where to send the user after they've completed the widget by setting the URL parameter in the `WidgetCompletionInfo` object. The child documentKey will be appended as a parameter to that URL if the sender is the same as the API key user.

Now that you know the child documentKey, you can call the `getDocumentInfo` method to get the latest status of the document.