

API documentation for SignSpace v2

Table of contents

- [Intro](#)
- [Authentication](#)
- [1. Create File\(s\)](#)
- [2. Create Signing Request](#)
- [Downloading Signed Files](#)
 - [Listing completed signing requests](#)
 - [Downloading files](#)
- [Cancel Signing Request](#)
- [Re-open Signing Request](#)
- [Edit Files of a Signing Request](#)
- [Enable SSN \(Social Security Number\) Verification](#)
- [Adding Metadata To a Signing Request](#)
- [Retention](#)
- [Content Removal](#)

Intro

This is a stripped-down version of the API documentation suited for the most basic integration case. In short, these are the steps in order to successfully create a signing request:

1. Authenticate yourself by using instructions provided here: [Authentication](#)
2. Create file or files in SignSpace by following instructions provided here: [Create File\(s\)](#). Store the created files' IDs internally for further use.
3. Create a signing request with the files created in Step 2 by following instructions provided in [Create Signing Request](#). Store the ID of the created signing request for further use.
4. Retrieve signed documents by following instructions from [Downloading Signed Files](#)

Authentication

Here are the respective URLs for partner sandbox environment and the production environment:

	Partner sandbox environment	Production environment
Cognito IdP URL	<code>https://signspace.staging-vastuugroup.fi/auth/token</code>	<code>https://app.signspace.com/auth/token</code>
Signspace API URL	<code>https://signspace.staging-vastuugroup.fi/api/v1/</code>	<code>https://app.signspace.com/api/v1/</code>

To retrieve an access token, make a POST request to identity provider URL similar to this:

```
1 curl -X POST --user $CLIENT_CREDENTIALS -d "grant_type=client_credentials" $IDENTITY_PROVIDER_URL
```

\$CLIENT_CREDENTIALS - the client ID and secret separated by a colon (<client_id>:<client_secret>)

Once you receive the token, use it in all requests like this:

```
1 curl -X GET -H "Authorization: Bearer $TOKEN" $URL
```

1. Create File(s)

Endpoint: */api/v1/files.create*

Description: Creates a file from Base64 encoding and returns the created object.

Allowed parameters:

Parameter	Description
acting_handle	Id of value of the acting handle
name	Name of the file
base64	Base64 encoding of the file
importance_order	Number of priority for displaying the file in the signing process. Higher the number, the earlier the file will be shown.

Note: Parameters in bold are mandatory.

Example request payload:

```
1 {
2   "acting_handle": "robot.email@domain.com",
3   "base64": "dGVzdCBkb2N1bWVudA==",
4   "name": "File name",
5   "importance_order": 1
6 }
```

cURL:

```
1 curl -XPOST -H "Authorization: Bearer $TOKEN" -H "Content-type: application/json" -d '{"acting_handle":
"robot.email@domain.com", "base64": "dGVzdCBkb2N1bWVudA==",
2 "name": "File name", "importance_order": 1}' 'https://signspace.staging-vastuugroup.fi/api/v1/files.create'
```

2. Create Signing Request

Endpoint: */api/v1/tasks.signingRequest.create*

Description: Creates a signing request task and returns the created object.

Allowed parameters:

Parameter	Description
-----------	-------------

acting_handle	ID or email of the API robot																
files	List of files IDs. Can be retrieved from the response payload of files.create (field name <code>_id</code>)																
participants	<p>List of participants/signatories objects:</p> <table border="1"> <thead> <tr> <th>Field</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>handle</td> <td>Email address of the recipient</td> </tr> <tr> <td>role</td> <td>Role of the participant. Use "signer" for signatory and "watcher" for CC.</td> </tr> <tr> <td>nin</td> <td>Signatory's social security number. Either nin or display_name are necessary for the user to be able to access the signature request</td> </tr> <tr> <td>display_name</td> <td>Full name of the signatory. Either nin or display_name are necessary for the user to be able to access the signature request</td> </tr> <tr> <td>forced_auth_method</td> <td> Signature method enforced for this signatory. Options are: <code>light_email_or_sms</code> - meaning that the user will sign via an SMS message (ultralight) or an email (light) <code>light_email</code> - meaning that the user will sign via email (light) <code>strong</code> - meaning that the user will have to strongly authenticate Note: if "is_protected" setting is enabled, users will have to strongly authenticate regardless </td> </tr> <tr> <td>invitation_methods</td> <td> List of methods to invite the users with. Options are: <code>sms</code> - Via SMS (if set, phone_number is mandatory to add) <code>email</code> - Via Email </td> </tr> <tr> <td>phone_number</td> <td>Phone number of the user. Example: +3580412345678</td> </tr> </tbody> </table> <p>Example for adding a signatory with email "signatory@email.com" which will be invited to sign by SMS:</p> <pre> 1 { 2 "handle": "signatory@email.com", 3 "role": "signer", 4 "nin": "12345-1", 5 "display_name": "Signatory Name", 6 "phone_number": "+3580412345678", 7 "forced_auth_method": "light_email_or_sms", 8 "invitation_methods": ["sms"] 9 }</pre> <p>Example for adding a signatory with email "signatory@email.com" which will be invited to sign by Email:</p> <pre> 1 { 2 "handle": "signatory@email.com", 3 "role": "signer", 4 "nin": "12345-1", 5 "display_name": "Signatory Name", 6 "phone_number": "+3580412345678", 7 "forced_auth_method": "light_email"</pre>	Field	Description	handle	Email address of the recipient	role	Role of the participant. Use "signer" for signatory and "watcher" for CC.	nin	Signatory's social security number. Either nin or display_name are necessary for the user to be able to access the signature request	display_name	Full name of the signatory. Either nin or display_name are necessary for the user to be able to access the signature request	forced_auth_method	Signature method enforced for this signatory. Options are: <code>light_email_or_sms</code> - meaning that the user will sign via an SMS message (ultralight) or an email (light) <code>light_email</code> - meaning that the user will sign via email (light) <code>strong</code> - meaning that the user will have to strongly authenticate Note: if "is_protected" setting is enabled, users will have to strongly authenticate regardless	invitation_methods	List of methods to invite the users with. Options are: <code>sms</code> - Via SMS (if set, phone_number is mandatory to add) <code>email</code> - Via Email	phone_number	Phone number of the user. Example: +3580412345678
Field	Description																
handle	Email address of the recipient																
role	Role of the participant. Use "signer" for signatory and "watcher" for CC.																
nin	Signatory's social security number. Either nin or display_name are necessary for the user to be able to access the signature request																
display_name	Full name of the signatory. Either nin or display_name are necessary for the user to be able to access the signature request																
forced_auth_method	Signature method enforced for this signatory. Options are: <code>light_email_or_sms</code> - meaning that the user will sign via an SMS message (ultralight) or an email (light) <code>light_email</code> - meaning that the user will sign via email (light) <code>strong</code> - meaning that the user will have to strongly authenticate Note: if "is_protected" setting is enabled, users will have to strongly authenticate regardless																
invitation_methods	List of methods to invite the users with. Options are: <code>sms</code> - Via SMS (if set, phone_number is mandatory to add) <code>email</code> - Via Email																
phone_number	Phone number of the user. Example: +3580412345678																

create_conversation	If set to "true", this will create a container for the signing request. For version v2, set this to True. This replaces the need for calling two more endpoints to fully create and send out the signing request.
participants_list_order_respected	If set to "true", you can set order in which the signatories will sign in
owners_handles	List of handles that will have owner privileges (acting_handle is added by default). To add group, use this format "group:63a199d56cd3aa8f6875e3b3" where 63a199d56cd3aa8f6875e3b3 is the group ID.
signature_strength	"light" - Signatory will be able to sign without any verifications (this is the default) "strong" - Signatory will perform one-time strong electronic identification during the signing process using Bank ID or Mobile ID
title	Title of signing request
description	Description of task
creator_handle	Can be passed if another person or group should be displayed as the signature request's creator. If the passed email is registered, that user's full name will be displayed.
creator_name	Can be passed when the creator is not registered. This name will be displayed in the UI as the creator's full name as well as in the email notifications.
company_name	Can be passed to configure which company name to display in signing flows and emails.
version	Version of signing request to use. Either "v1" or "v2". For version 2 use "v2"
recipient_language	Set the language in which users will receive emails regarding the signing request. Options: "en", "fi", "sv". Default: "en"
due_date	ISO format date of due date. After passing the date, signing request is put in "OVERDUE" state. You can control whether signatories can sign in that state with "overdue_signing_disabled" parameter
overdue_signing_disabled	If set to "true", signing after due_date is not possible. Default is "false"
invitation_type	"default_signing" - Default setting. All signatories gain access to the space of the signing request. "force_swr" - If set, both registered and unregistered signatories will be forced to use the signing without registration flow. Signatories will not gain access to the space of the signing request (except for the creator of the signing request)
cannot_sign_disabled	If set to "true", users will not be able to decline signing. Default is "false"
signing_discussion_disabled	If set to "true", users will not be able to post any messages to the conversations related to the signing request. Default is "false"

custom_customer_reference	A free-text field where you can store information about your internal customer. Visible in exports of signature reports								
is_protected	If set to "true", users will have to verify their identity to view the signature request. The social security number has to match with the one saved on the participant data, otherwise user will not be permitted to read the request.								
retention_settings	<p>Object (Retention description):</p> <table border="1"> <thead> <tr> <th>Field</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>delete_signed_content_after_months</td> <td>Time (months) Signed content will be deleted after. Optional</td> </tr> <tr> <td>delete_unsigned_content_after_months</td> <td>Time (months) Unsigned content will be deleted after. Optional</td> </tr> <tr> <td>delete_conversations_and_tasks_after_months</td> <td>Time (months) Other content will be deleted after. Optional</td> </tr> </tbody> </table> <p>Example:</p> <pre> 1 { 2 "delete_unsigned_content_after_months": 12, 3 "delete_signed_content_after_months": 84, 4 "delete_conversations_and_tasks_after_months": 120 5 }</pre>	Field	Description	delete_signed_content_after_months	Time (months) Signed content will be deleted after. Optional	delete_unsigned_content_after_months	Time (months) Unsigned content will be deleted after. Optional	delete_conversations_and_tasks_after_months	Time (months) Other content will be deleted after. Optional
Field	Description								
delete_signed_content_after_months	Time (months) Signed content will be deleted after. Optional								
delete_unsigned_content_after_months	Time (months) Unsigned content will be deleted after. Optional								
delete_conversations_and_tasks_after_months	Time (months) Other content will be deleted after. Optional								

Note: Parameters in bold are mandatory.

Example request payload:

```

1 {
2   "files": ["62a85fd06cd3aa6e70d1af2d"],
3   "title": "Signing request title",
4   "description": "Signing request description",
5   "acting_handle": "robot.email@domain.com",
6   "owners_handles": ["group:1234567890", "email@domain.com"],
7   "is_protected": true,
8   "recipient_language": "fi",
9   "create_conversation": true,
10  "version": "v2",
11  "participants": [{
12    "handle": "signatory@email.com",
13    "role": "signer",
14    "nin": "12345-1",
15    "display_name": "Signatory Name",
16    "phone_number": "+3580412345678",
17    "forced_auth_method": "light_email_or_sms",
18    "invitation_methods": ["sms"]
19  }],
20  "retention_settings": {
21    "delete_unsigned_content_after_months": 12,
22    "delete_signed_content_after_months": 60
23  }
24 }
```

cURL:

```

1 curl -XPOST -H "Authorization: Bearer $TOKEN" -H "Content-type: application/json" -d '{"files":
["62a85fd06cd3aa6e70d1af2d"], "title": "Signing request title", "description": "Signing request description",
```

```
"acting_handle": "robot.email@domain.com", "owners_handles": ["group:1234567890", "email@domain.com"],
"is_protected": true, "recipient_language": "fi", "create_conversation": true, "participants": [{ "handle":
"signatory@email.com", "role": "signer", "nin": "12345-1", "display_name": "Signatory Name", "phone_number":
"+3580412345678", "forced_auth_method": "light_email_or_sms", "invitation_methods": ["sms"] }],
"retention_settings": { "delete_unsigned_content_after_months": 12, "delete_signed_content_after_months": 60}}'
'https://signspace.staging-vastuugroup.fi/api/v1/tasks.signingRequest.create'
```

Downloading Signed Files

Summary: In order to retrieve the signed documents you will need to fetch all completed signing requests by either using endpoint `/api/v1/tasks.list` or `/api/v1/tasks.get` (if you have the IDs of the signing requests). Both endpoints return a list of signing requests. You can retrieve the IDs of signed files by accessing the field **"signed_files"** of these signing request objects. Once the IDs are retrieved, files can be downloaded via `/api/v1/files.download` endpoint. More information about how to use the endpoints below:

Listing completed signing requests

Endpoint: `/api/v1/tasks.list` (GET)

Description: Returns all signing requests available.

Allowed parameters:

Parameter name	Options	Description
status	"complete" "in_progress" "open" "on_hold" "cancelled"	Filter by status of the signing requests
type	"signing_request"	Filter by type of task. For this case it should always be "signing_request"
limit	Default: 20	Limit the maximum number of signing requests returned

Example cURL:

```
1 curl -XGET -H "Authorization: Bearer $TOKEN" 'https://signspace.staging-vastuugroup.fi/api/v1/tasks.list?
type=signing_request&status=complete&limit=100'
```

Endpoint: `/api/v1/tasks.get` (GET)

Scopes needed: `tasks.get`

Description: Returns a list of signing requests filtered by given IDs.

Example cURL:

```
1 curl -XGET -H "Authorization: Bearer $TOKEN" 'https://signspace.staging-vastuugroup.fi/api/v1/tasks.get?
id=62a85fc66cd3aa6e70d1af2c&id=62a85fd06cd3aa6e70d1af2d'
```

Downloading files

Endpoint: `/api/v1/files.download`

Allowed methods: GET

Description: Downloads file that matches the given ID. Also scans the file for viruses if the file has not been scanned in the last 6 hours.

Allowed parameters:

Parameter	Description
id	Id or a list of IDs of file to download

cURL:

```
1 curl -XGET -H "Authorization: Bearer $TOKEN" 'https://signspace.staging-vastuugroup.fi/api/v1/files.download?id=62a85fc66cd3aa6e70d1af2c&id=62a85fd06cd3aa6e70d1af2d'
```

Cancel Signing Request

Endpoint: `/api/v1/tasks.cancel`

Allowed methods: POST

Description: Marks the task as cancelled, disabling participants from performing actions to it.

Allowed parameters:

acting_handle	Value or id of handle that is initializing the signing
task	ID of the signing request

Example payload:

```
1 {
2   "acting_handle": "robot.email@domain.com",
3   "task": "63a198e76cd3aa8f6875e3b2"
4 }
```

cURL:

```
1 curl -XPOST -H "Authorization: Bearer token" -H "Content-type: application/json" -d '{"acting_handle": "robot.email@domain.com", "task": "63a198e76cd3aa8f6875e3b2"}' 'https://signspace.staging-vastuugroup.fi/api/v1/tasks.cancel'
```

Re-open Signing Request

Endpoint: `/api/v1/tasks.open`

Allowed methods: POST

Description: Re-opens the signing request after it was cancelled or put on hold. After the signing request is re-opened, all the pending signatories can sign the attached documents.

Allowed parameters:

acting_handle	Value or id of handle that is initializing the signing
task	ID of the signing request

Example payload:

```
1 {
2   "acting_handle": "robot.email@domain.com",
3   "task": "63a198e76cd3aa8f6875e3b2"
4 }
```

cURL:

```
1 curl -XPOST -H "Authorization: Bearer token" -H "Content-type: application/json" -d '{"acting_handle":
"robot.email@domain.com", "task": "63a198e76cd3aa8f6875e3b2"}' 'https://signspace.staging-
vastuugroup.fi/api/v1/tasks.open'
```

Edit Files of a Signing Request

Endpoint: `/api/v1/tasks.signingRequest.update`

Allowed methods: POST

Description: Endpoint for editing signing request's parameters, including files. The endpoint will set the new list of files on the signing request. This means that previous files will no longer be signable if their IDs are not in the files list.

Allowed parameters:

acting_handl e	Value or id of handle that is initializing the signing
task	ID of the signing request
files	List of file ids

Example payload:

```
1 {
2   "acting_handle": "robot.email@domain.com",
3   "task": "63a198e76cd3aa8f6875e3b2",
4   "files": ["63a19b546cd3aa8f6875e3b4", "63a19b566cd3aa8f6875e3b5"]
5 }
```

cURL:

```
1 curl -XPOST -H "Authorization: Bearer token" -H "Content-type: application/json" -d '{"acting_handle":
"robot.email@domain.com", "task": "63a198e76cd3aa8f6875e3b2", "files": ["63a19b546cd3aa8f6875e3b4",
"63a19b566cd3aa8f6875e3b5"]}' 'https://signspace.staging-vastuugroup.fi/api/v1/tasks.signingRequest.update'
```

Enable SSN (Social Security Number) Verification

In order to enable verification of SSN for signatories, the request payload of [/api/v1/tasks.signingRequest.create](#) needs to be appended with **"is_protected: true"**

If protection is enabled, the participants models can now also be appended with property **"nin"**. Example payload:

Example request payload:

```
1 {
2   ...
3   "is_protected": true,
4   "participants": [
5     {
6       "handle": "signatory2@domain.com",
7       "role": "signer",
8       "nin": "234242-12"
9     },
10    {
11      "handle": "signatory1@domain.com",
12      "role": "signer",
13      "display_name": "Name Surname",
14      "nin": "123456-12"
15    }
16  ],
17 }
```

If **nin** is set on the participant model with signer role, the signatory will have to strongly verify their identity. If the SSN returned from the identity provider matches, the user is able to access the documents and sign.

If **display_name** is also set on the participant model with protection enabled, the name will also be compared against their full name returned from the identity provider. If at least two words in the full name match, the verification is approved.

Adding Metadata To a Signing Request

Metadata is supported on signing requests on SignSpace. In order to take it into use, the [metadata fields](#) and [metadata templates](#) have to be created in the SignSpace UI by an organisation admin. Once done, the IDs of the fields and templates have to be stored and mapped on your end.

In order to add metadata on a signing request, append the request payload of [/api/v1/tasks.signingRequest.create](#) as follows:

Example payload:

```
1 {
2   ...
3   "metadata_template": "6659c53f6cd3aa12a84e08f3", # ID of the metadata template that was created in the
SignSpace UI
4   "metadata": {
5     "6659c5826cd3aa12a84e08f4": {"value": "Example A"}, # Key is an ID of the metadata field created in the
UI. Value is passed as {"value": value}
6     "6659c5826cd3aa12a84e08f5": {"value": "Example B"},
7   }
8 }
```

Retention

Retention can be set on signing requests on SignSpace.

Retention object can consist of:

Field	Description
delete_signed_content_after_months	Time (months) Signed content will be deleted after. Optional
delete_unsigned_content_after_months	Time (months) Unsigned content will be deleted after. Optional
delete_conversations_and_tasks_after_months	Time (months) Other content will be deleted after. Optional

Retention can be taken into use via 2 ways:

- [/api/v1/tasks.signingRequest.create](#) - Including `retention_settings` in the JSON body
- [/api/v1/conversations.setRetentionManually](#) - Setting retention manually for specific tasks or conversations

Endpoint: [/api/v1/conversations.setRetentionManually](#)

Allowed methods: POST

Description: Updates retention for specific task or conversion.

Allowed parameters:

acting_handle	Value or id of handle that is initializing the signing
retention_settings	retention_settings object
task	ID of the signing request (Conversation or Task must be provided)
conversation	ID of the conversation (Conversation or Task must be provided)

Example payload:

```
1  {
2    "acting_handle": "robot.email@domain.com",
3    "task": "<Id of signing request task>",
4    "conversation": "<Id of conversation>",
5    "retention_settings": {
6      "delete_unsigned_content_after_months": 12,
7      "delete_signed_content_after_months": 84,
8      "delete_conversations_and_tasks_after_months": 120
9    }
10 }
```

Content Removal

Endpoint: [/api/v1/conversations.markDeleted](#)

Allowed methods: POST

Description: Marks one or more conversations for deletion. 14 days of grace period is given, during this period marked content can be restored. After the grace period is over, content will be fully removed.

Allowed parameters:

acting_handle	Value or id of handle that is initializing the signing
conversations	List of conversation IDs. Greater or equal to one
email_recipients_handles	Handles to send email. Optional; only meaningful for single conversation
message_to_recipients	Message to send. Optional; only meaningful for single conversation

Example payloads:

```
1 {  
2   "acting_handle": "robot.email@domain.com",  
3   "conversations": ["507f1f77bcf86cd799439011"],  
4   "message_to_recipients": "This space will be deleted in 14 days.",  
5   "email_recipients_handles": ["colleague-handle", "other-handle"]  
6 }
```