Set up the IndiVideo integration

Content updated: 10/21/2026

This article is designated for administrators.

About

The **IndiVideo** integration lets you add personalized video to your own application, using web components and REST APIs. To set it up, you choose an environment, authenticate, load and configure the component library, and add a video player to your page.

Before you start

Make sure you have:

- An IndiVideo API username and password, or an SSO setup configured for your organization.
- The component library version you want to use.
- A video attachment code and language for the video you want to display.

If you need any of these, open a Kaltura Support ticket.

Step 1: Choose an environment

IndiVideo provides separate production and sandbox environments. Use the API and component library for the same environment.

- **Production API:** `https://individeo.com/rest/v1/`
- **Sandbox API:** `https://sb.individeo.com/rest/v1/`
- **Production component library:** `https://cdn.individeo.com/components/iv/{version}/components.js`
- **Sandbox component library:** `https://cdn.individeo.com/components/sb/{version}/components.js`

Replace `{version}` with a fixed version, in the form `{major}/{major}_{minor}_{patch}`, or with `edge`, which always loads the latest stable version. In production, use a fixed version, so library updates don't change your integration without warning. Use `edge` in the sandbox or for testing.

Test your implementation in the sandbox before you use it in production.

Copyright © 2025 Kaltura Inc. All Rights Reserved. Designated trademarks and brands are the property of their respective owners. Use of this document constitutes acceptance of the Kaltura Terms of Use and Privacy Policy.

Step 2: Authenticate

Authenticate before you call protected REST APIs or initialize components that require a logged-in user. IndiVideo supports plugin authentication and SAML single sign-on (SSO).

Use plugin authentication

Send a `POST` request with URL-encoded form data to the plugin authentication endpoint for your environment:

- **Production:** `https://individeo.com/rest/v1/plugin/authenticate`
- **Sandbox:** `https://sb.individeo.com/rest/v1/plugin/authenticate`

The request uses your IndiVideo API username and password, which are provided separately. Run the request from your application server; don't include the API password in browser code.

```
curl --request POST "${INDIVIDEO_API_BASE}/plugin/authenticate" \
  --header "Content-Type: application/x-www-form-urlencoded" \
  --data-urlencode "username=${INDIVIDEO_USERNAME}" \
  --data-urlencode "password=${INDIVIDEO_PASSWORD}"
```

Set `INDIVIDEO_API_BASE` to the production or sandbox API base URL. From a successful response, use `token.authToken` as the authentication token and `user.username` as the username.

The token expires after the number of seconds in `token.ttl`, which is 24 hours by default. IndiVideo has no refresh endpoint. When the token expires, call `plugin/authenticate` again from your server, and pass the new token to `window.IndivideoSettings()`. The page doesn't need to reload.

Use SAML SSO

If users log in through your organization's identity provider, open a Kaltura Support ticket to set up SAML SSO before you use it.

Step 3: Load the component library

Add the library URL for your environment to your page. This example loads the production library:

```
<script type="module" src="https://cdn.individeo.com/components/iv/{version}/components.js"></script>
```

For the sandbox, use the component library URL that contains `/components/sb/`. Load the library before calling `window.IndivideoSettings()`.

Copyright © 2025 Kaltura Inc. All Rights Reserved. Designated trademarks and brands are the property of their respective owners. Use of this document constitutes acceptance of the Kaltura Terms of Use and Privacy Policy.

Step 4: Configure the components

After the library has loaded and your application has the authentication values, pass the token and username to `window.IndivideoSettings()`:

```
window.IndivideoSettings({
  authToken: authData.token.authToken,
  username: authData.user.username,
  lang: 'en'
});
```

Set `lang` to `en` or `fr`. For authenticated REST API calls, send the same username and token in the `username` and `access-token` headers.

```
Content-Type: application/json; charset=UTF-8
Accept: application/json; charset=UTF-8
username: <username>
access-token: <authToken>
```

To check that authentication works, call `GET /rest/v1/user/me` with these headers. Use the production or sandbox API base URL that matches your component library.

Step 5: Add a video player

Add the `<iv-video-player>` component to your page. Use the video's `attachmentCode` and language when possible. Video usage isn't tracked when you use a `mediaCode`.

```
<iv-video-player
  attachmentCode="<attachmentCode>"
  lang="en-CA">
</iv-video-player>
```

Replace `<attachmentCode>` with the code for the video and environment you want to display.

Listen for the component's `error` event so your application can report playback errors:

```
document.querySelector('iv-video-player').addEventListener('error', (event) => {
  console.error('IndiVideo player error:', event.detail);
});
```

Step 6: Check the integration

- Confirm that your application uses the same environment for the API and component library.
- Confirm that `window.IndivideoSettings()` receives the authenticated username and token.
- Confirm that the player loads the expected video.

Copyright © 2025 Kaltura Inc. All Rights Reserved. Designated trademarks and brands are the property of their respective owners. Use of this document constitutes acceptance of the Kaltura Terms of Use and Privacy Policy.

- Check the browser console and the component's `error` event for errors.

For local development, IndiVideo allows calls from localhost port `4201`. Calls from other localhost ports are blocked. To allow another origin, open a Kaltura Support ticket.

Next steps

To record, edit, and save videos, see Record and save videos with IndiVideo components.

Copyright © 2025 Kaltura Inc. All Rights Reserved. Designated trademarks and brands are the property of their respective owners. Use of this document constitutes acceptance of the Kaltura Terms of Use and Privacy Policy.