Security module

Last Modified on 11/18/2024 2:49 pm IST

This article is designated for administrators.

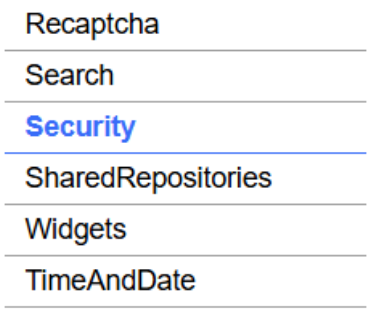
About

The Security module in Kaltura's Video Portal helps administrators implement advanced security measures to protect the platform. Key features include enforcing HTTPS connections, restricting playback to specific IPs, and configuring HTTP headers like Content-Security-Policy, Permissions-Policy, and Cross-Origin policies.

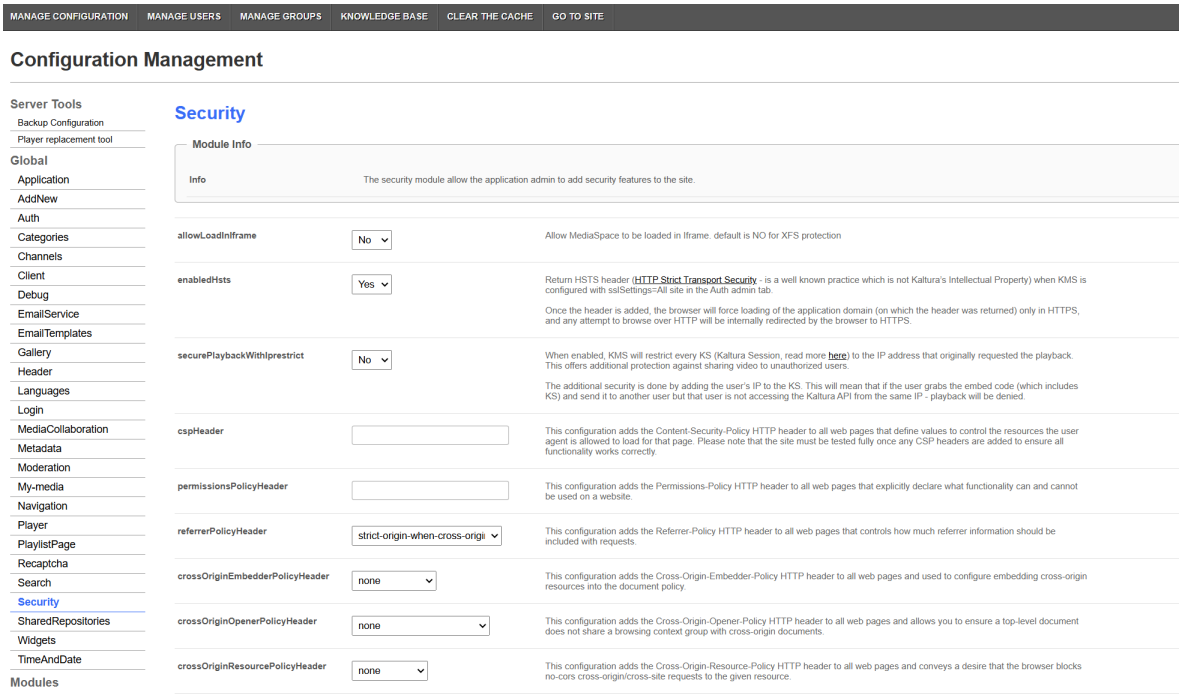
These tools ensure secure access, control resource loading, and protect against unauthorized sharing and other vulnerabilities.

Configure

1. Go to your Configuration Management console and click on the **Security** module.
   You can also navigate to it directly using a link:
   https://{your_KMS_URL}/admin/config/tab/security.



   The Security page displays.



2. Configure the following:

   allowLoadInIframe - Allow MediaSpace to be loaded in Iframe. The default is NO for XFS.

   enabledHsts - Return HSTS header. The HSTS header (HTTP Strict Transport Security) is a widely recognized security practice (not proprietary to Kaltura) used to enhance web application security. In Kaltura's Video Portal, this header is automatically returned when the **sslSettings** are configured as **All Site** in the Auth admin tab.

   When the HSTS header is applied, the browser enforces HTTPS for the application domain where the header is set. Any attempt to access the site over HTTP will be internally redirected by the browser to HTTPS, ensuring secure connections at all times.

   securePlaybackWithIprestrict - When enabled, this feature restricts every Kaltura Session (read more here) to the IP address that originally requested the playback, providing an additional layer of protection against unauthorized sharing.

   The user's IP address is embedded into the KS, ensuring that if someone copies the embed code (which includes the KS) and shares it with another user, playback will be denied unless the second user accesses the Kaltura API from the same IP address.

   cspHeader - This configuration adds the Content-Security-Policy (CSP) HTTP header to all web pages, specifying which resources the user agent is allowed to load for each page. It is important to thoroughly test the site after implementing any CSP

headers to ensure all functionality operates as expected.

permissionsPolicyHeader - This configuration adds the Permissions-Policy HTTP header to all web pages, explicitly specifying which functionalities are allowed or restricted on the website.

referrerPolicyHeader - This configuration adds the Referrer-Policy HTTP header to all web pages that controls how much referrer information should be included with requests. Choose from the following options:

- no-referrer
- no-referrer-when-downgrade
- origin
- origin-when-cross-origin
- same-origin
- strict-origin
- strict-origin-when-cross-origin
- unsafe-url

crossOriginEmbedderPolicyHeader - This configuration adds the Cross-Origin-Embedder-Policy HTTP header to all web pages and used to configure embedding cross-origin resources into the document policy.

Choose from the following options:

- none
- unsafe-none
- require-corp
- credentialless

crossOriginOpenerPolicyHeader - This configuration adds the Cross-Origin-Opener-Policy HTTP header to all web pages and allows you to ensure a top-level document does not share a browsing context group with cross-origin documents.

Choose from the following options:

- none
- unsafe-none
- same-origin-allow-popups
- same-origin

crossOriginResourcePolicyHeader - This configuration adds the Cross-Origin-Resource-Policy HTTP header to all web pages instructing the browser to block no-cors

cross-origin or cross-site requests to the specified resource.

Choose from the following options:

- none
- same-site
- cross-origin
- same-origin

- When you have finished configuring the module, click **Save**.