Security module

Content updated: 10/30/2026

This article is designated for administrators.

About

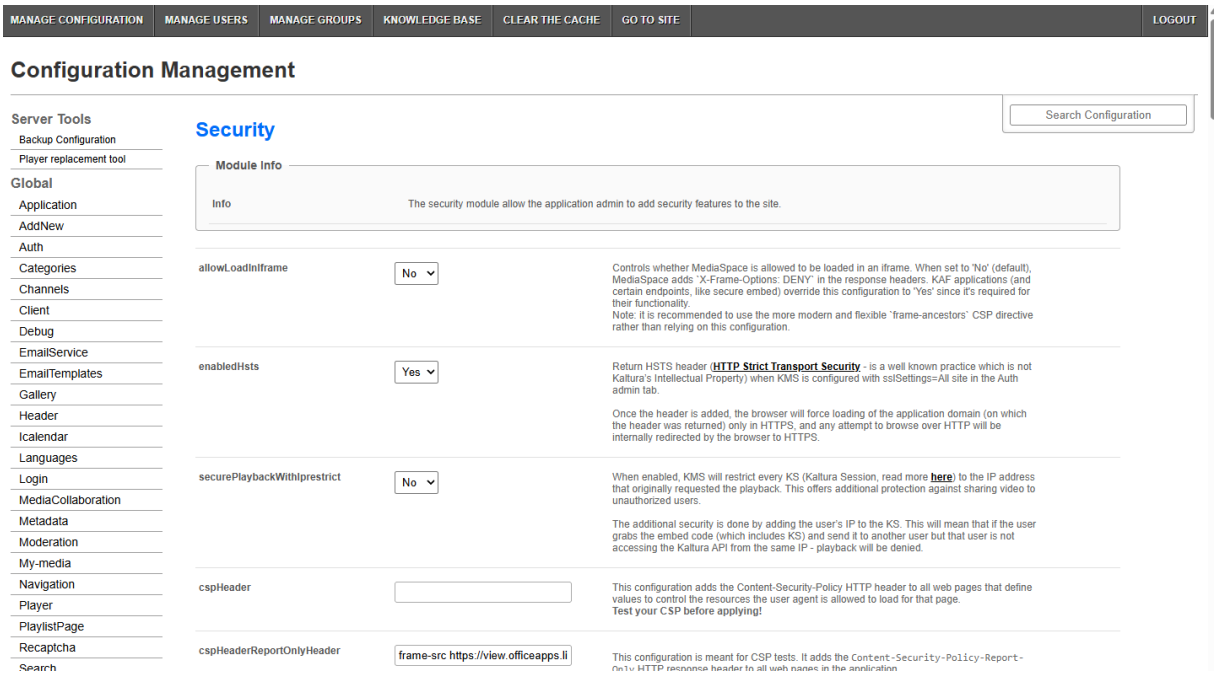
The **Security** module lets administrators configure security settings for **Content Hubs** and **legacy Video Portal** sites, and for **Kaltura Application Framework (KAF)** integrations, such as **LMS** integrations.

These settings include enforcing HTTPS connections, restricting playback to specific IP addresses, and configuring HTTP security headers. HTTP security headers help control how browsers load resources, share information, and interact with content from other origins.

Configure the module

1. Go to your Configuration Management console, locate the **Security module** in the left panel, then select it. You can also navigate directly to: `https://{your_site_url}/admin/config/tab/security`

The **Security** page displays.



2. Configure the following settings:

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.

allowLoadInIframe - Allows the site to be displayed inside another website's iframe (embedded). The default is **No**. Enable this setting only when the site needs to be embedded in another website or application.

enabledHsts - Returns the HTTP Strict Transport Security (HSTS) header, which tells browsers to always use HTTPS when accessing the application domain.

This header is returned automatically when `sslSettings` is set to **All Site** in the **Auth** module.

securePlaybackWithIprestrict - Restricts each Kaltura Session (KS) to the IP address from which playback was requested, providing additional protection against unauthorized sharing. For more information, see How to create a Kaltura Session.

The user's IP address is included in the KS. If someone copies and shares an embed code containing that KS, playback is denied when the recipient accesses the Kaltura API from a different IP address.

Content Security Policy (CSP) settings

A **Content Security Policy (CSP)** tells browsers which sources the site is allowed to load resources from, such as scripts, styles, images, fonts, media, and embedded content. This helps protect the site against cross-site scripting and other code-injection attacks.

> **Start with `cspHeaderReportOnlyHeader`, not `cspHeader`.** Add and test your policy in `cspHeaderReportOnlyHeader` first. Copy it into `cspHeader` only after testing.

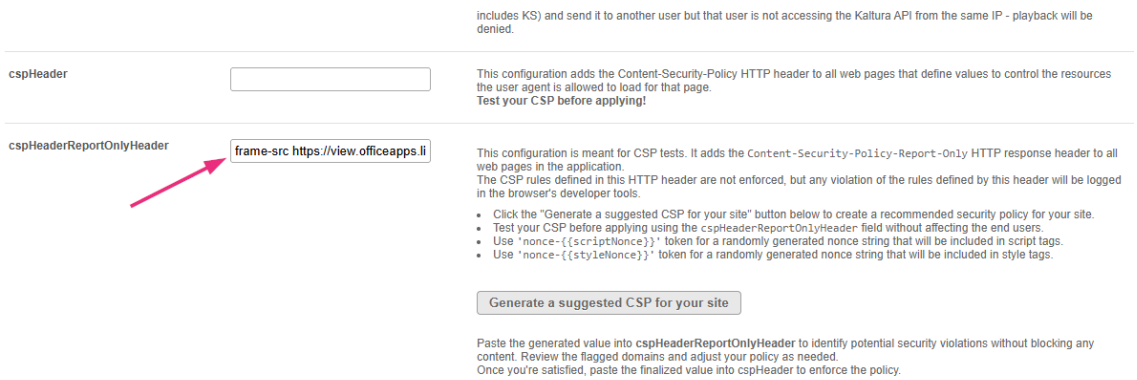
cspHeader - Adds the `Content-Security-Policy` HTTP header to all web pages in the application. The policy controls which resources the browser can load for each page. If `adminCspHeader` is empty, this policy also applies to the Configuration Management console.

Leave this field empty until you've tested your policy in `cspHeaderReportOnlyHeader`, the next field. A policy in `cspHeader` is applied as soon as you save, and it can block content on your site.

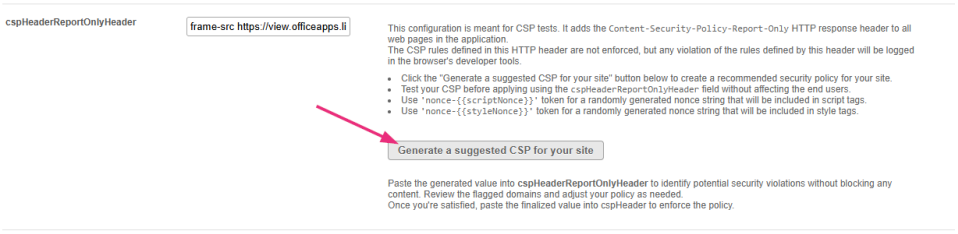
If `cspHeader` is empty, administrators see a security alert once per login session in the Configuration Management console. Kaltura recommends configuring a CSP header to meet standard security requirements. The alert doesn't mean there's been a security breach. Administrators can dismiss it for the current session, but it appears again during subsequent sessions until `cspHeader` is populated.

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.

cspHeaderReportOnlyHeader - Adds the `Content-Security-Policy-Report-Only` HTTP response header to all web pages in the application. The policy isn't enforced, but violations are logged in the browser's developer tools. Use this field to test your policy before adding it to `cspHeader`.



To get a suggested policy to start with, select **Generate a suggested CSP for your site**, under the `cspHeaderReportOnlyHeader` field. The suggested policy is based on your site's configuration and enabled modules.



adminCspHeader - Adds an enforced Content-Security-Policy HTTP header specifically to the Configuration Management console. When this field is populated, it overrides `cspHeader` for the console. If it is empty, the console uses the policy configured in `cspHeader`.

Leave this field empty until you've tested your policy in `adminCspHeaderReportOnlyHeader`. A policy in this field can block access to the Configuration Management console.

adminCspHeaderReportOnlyHeader - Adds a Content-Security-Policy-Report-Only HTTP header specifically to the Configuration Management console. The policy isn't enforced, but violations are logged in the browser's developer tools.

Enter only the policy directives in the CSP fields. Don't include the `Content-Security-Policy:` or `Content-Security-Policy-Report-Only:` header name.

Your policy depends on the resources, integrations, and customizations your site uses, so test your site thoroughly whenever you add or change a policy. For more

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.

information, see:

- Configure a Content Security Policy (CSP) - How to generate, test, and apply a policy.
- Troubleshoot Content Security Policy (CSP) messages in the browser Console - How to read and fix the messages you see while you test.
- Content Security Policy (CSP) directives and sources - What each directive and source in your policy means.

Additional HTTP security headers

permissionsPolicyHeader - Adds the Permissions-Policy HTTP header, which turns off specific browser features (such as camera, microphone, or location access) for the site, even if a visitor's browser would otherwise allow them.

referrerPolicyHeader - Adds the Referrer-Policy HTTP header, which controls how much referrer information is included with requests. The default is `strict-origin-when-cross-origin`.

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 - Adds the Cross-Origin-Embedder-Policy HTTP header, which controls whether the page can load cross-origin resources.

Choose from the following options:

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

crossOriginOpenerPolicyHeader - Adds the Cross-Origin-Opener-Policy HTTP header, which controls whether a top-level document shares a browsing context group with cross-origin documents.

Choose from the following options:

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.

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

crossOriginResourcePolicyHeader - Adds the Cross-Origin-Resource-Policy HTTP header, which instructs browsers to block certain `no-cors` cross-origin or cross-site requests.

Choose from the following options:

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

Additional security settings

useTrueClientIp - Determines the user's IP address using a trusted HTTP header configured by Kaltura DevOps or through your CDN.

- **Enabled** - Uses the configured trusted header instead of potentially spoofed `X-Forwarded-For` values.
- **Disabled** - Uses standard IP detection, such as `REMOTE_ADDR`.

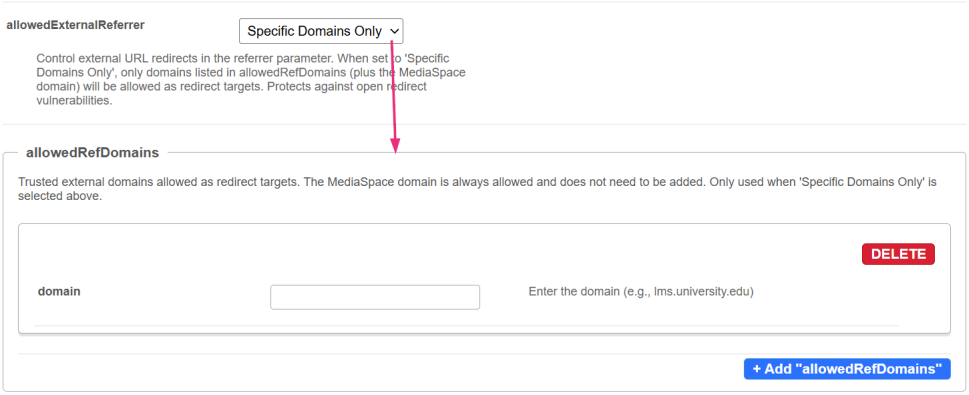
Enable this setting only when the trusted header has been properly configured.

allowedExternalReferrer - Controls which external domains can be used as redirect destinations in the `referrer` parameter. This helps protect the site against open-redirect vulnerabilities.

- **All Domains** - Allows redirects to any external domain.
- **Specific Domains Only** - Allows redirects only to the site domain and the trusted external domains configured in `allowedRefDomains`.

When you select **Specific Domains Only**, the `allowedRefDomains` field displays. The site domain is always allowed and doesn't need to be added.

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.



To add an allowed domain:

1. Select **Add "allowedRefDomains"**.
2. In the **domain** field, enter the trusted domain, for example, `lms.university.edu`.
3. Repeat these steps for each additional domain you want to allow.

hideVersionEndpoint - Controls whether the `/version` endpoint is publicly accessible.

- **Disabled** - Allows access to the endpoint.
- **Enabled** - Returns a `403` response and prevents unauthenticated users from viewing the application version and build number.

enableIdleTimeout - Automatically signs users out after a specified period of inactivity. Before they are signed out, users receive a warning and can choose to remain signed in.

- **Off** - Disables automatic sign-out due to inactivity.
- **On** - Enables automatic sign-out according to the values configured in `idleTimeoutDuration` and `warningPeriod`.

idleTimeoutDuration - Sets the number of minutes a user can remain inactive before being signed out automatically. Enter a value from `5` to `1440` minutes (24 hours). This setting applies only when `enableIdleTimeout` is enabled.

warningPeriod - Sets how many minutes before the idle timeout users are warned that they will be signed out. The value must be less than `idleTimeoutDuration`. This setting applies only when `enableIdleTimeout` is enabled.

3. When you have finished configuring the module, select **Save**.

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.