

User Guide

Video Pro for Magento 2.x — Extmag_Videopro

2026-05-23

- 1 About the module
- 2 Admin navigation
- 3 Configuration
 - 3.1 Allowed formats
 - 3.2 Player restrictions
- 4 Uploading video files
 - 4.1 From a product edit page
 - 4.2 From a CMS page, block or Page Builder content
 - 4.3 File browser — Upload Videos
- 5 Selecting a previously uploaded video
- 6 Editing the embedded `<video>` tag
- 7 YouTube and Vimeo embedding
- 8 Storefront behavior
- 9 Troubleshooting
 - 9.1 *Upload Videos* row missing
 - 9.2 Uploaded file rejected with *Disallowed file type*
 - 9.3 Video uploaded but does not play on the storefront
 - 9.4 Picture-In-Picture still appears after enabling *Disable Picture In Picture*
 - 9.5 Large uploads fail silently or with HTTP 413
 - 9.6 *Insert Video* button missing from TinyMCE
 - 9.7 Hyva theme — gallery thumbnail shows but player does not load

1 About the module

Video Pro lets you host product and content videos on your own Magento store. Instead of embedding YouTube or Vimeo for every product video, you upload the file once through the standard WYSIWYG media gallery and reuse it across product galleries, CMS pages, blocks, Page Builder content, category descriptions and widgets.

The module covers four jobs:

1. Restores the **Upload Videos** row in the WYSIWYG file browser so video files can be uploaded the same way as images. Includes a dedicated fix for the changed upload flow on **Magento 2.4.7+**.
2. Wires self-hosted videos into the product *Images And Videos* fieldset so a custom file can be a gallery item alongside images.
3. Adds an **Insert Video** button to the TinyMCE editor so videos can be embedded in any CMS / Page Builder content.
4. Applies HTML5 player restrictions (`controlsList` , `disablePictureInPicture`) globally from a single configuration section.

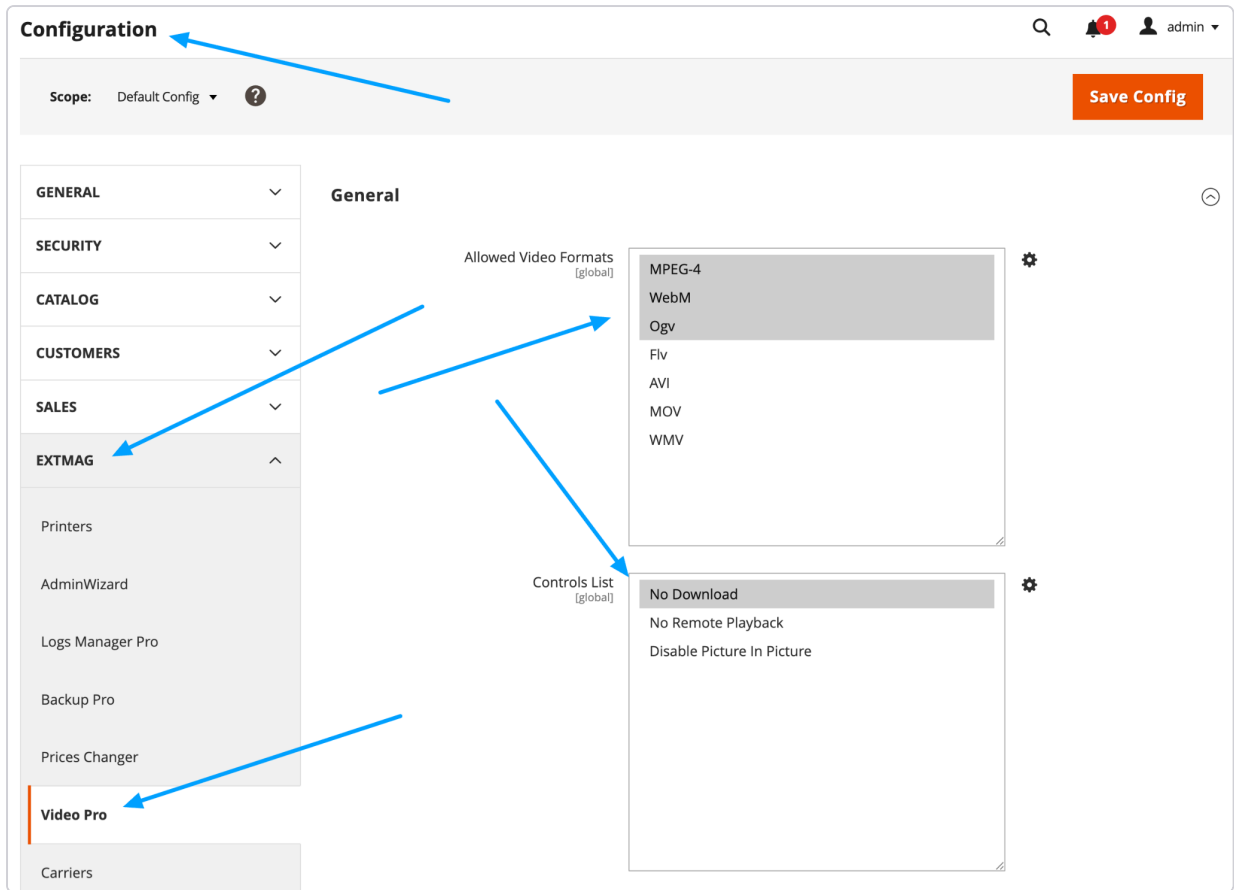
2 Admin navigation

Path	Purpose
Stores → Configuration → Extmag → Video Pro → General	Allowed formats + player restrictions.
Product edit → Images And Videos	Per-product video gallery item.
CMS / Block / Page Builder editor → <i>Insert Video</i> button	Embed a video in any rich-text content.

ACL resource: `Extmag_Videopro::config` (nested under `Extmag_Core::config`). Grant or revoke per admin role.

3 Configuration

Stores → Configuration → Extmag → Video Pro → General .



System configuration — Video Pro general section

Field	Default	Notes
Allowed Video Formats	mp4, webm, ogv	Multiselect — controls which file extensions the WYSIWYG file browser will accept. MP4 + WebM is the safe default.
Controls List	empty	Multiselect — enables HTML5 player restrictions. See Player restrictions .

Save the configuration after editing. Allowed formats take effect on the next file browser open (no cache flush required); controls list changes require a `bin/magento cache:flush` because the storefront `<video>` tag attributes are written from a cached layout block.

3.1 Allowed formats

Extension	MIME type	Browser playback
mp4	video/mp4	Universal (H.264 / AAC).
webm	video/webm	Chromium + Firefox + Safari 14+.
ogv	video/ogv / ogg	Firefox + Chromium. Niche.
flv	video/x-flv	No native HTML5 support — legacy / archive only.

Extension	MIME type	Browser playback
avi	video/x-msvideo	No native HTML5 support — legacy / archive only.
mov	video/quicktime	Safari only out of the box.
wmv	video/x-ms-wmv	No native HTML5 support — legacy / archive only.

Enable formats your storefront audience can actually play. `mp4` + `webm` covers ~100% of modern browsers; the rest are present so existing assets can be uploaded for reference, but they will not play in the storefront player without a third-party decoder.

3.2 Player restrictions

The *Controls List* multiselect maps directly onto the HTML5 `controlsList` and `disablePictureInPicture` attributes:

Token	Effect on the storefront player
<code>nodownload</code>	Removes the download item from the three-dot menu. (Chromium.)
<code>noremoteplayback</code>	Hides the Cast / AirPlay button. (Chromium + Safari.)
<code>disablepictureinpicture</code>	Disables PiP — the picture-in-picture button is removed and the API is rejected.

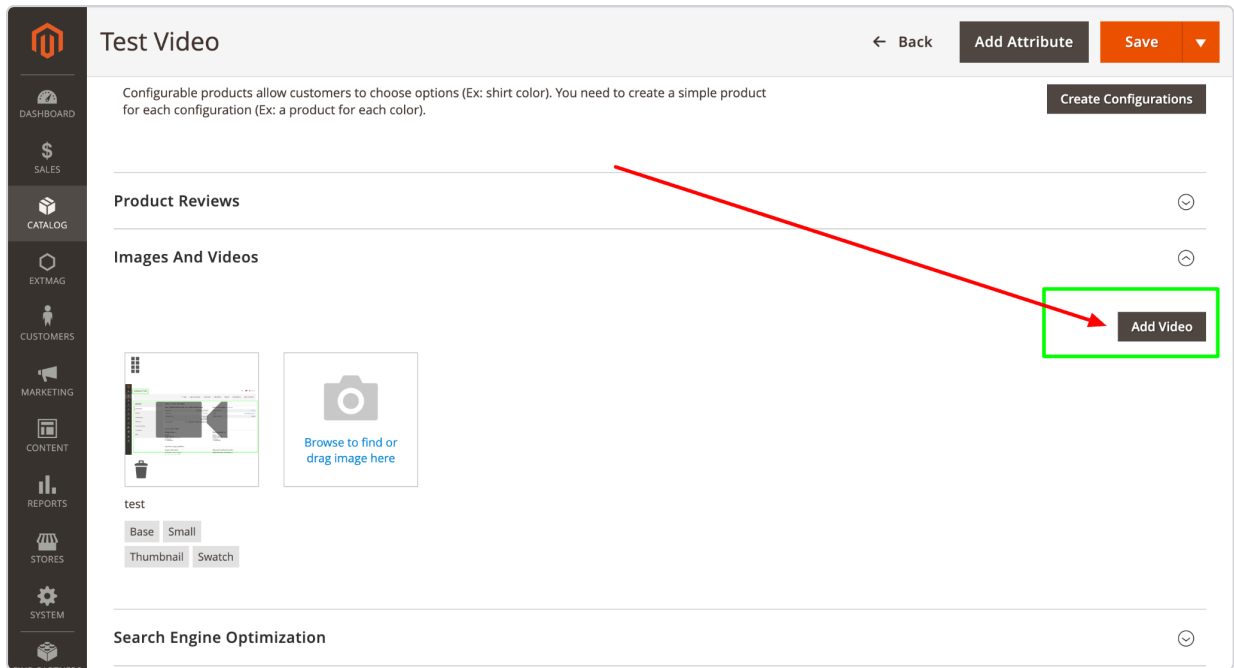
Tokens are emitted on every `<video controls>` element rendered by the storefront (product gallery + WYSIWYG-embedded videos). They are advisory hints to the browser — the storefront cannot prevent a determined user from saving the file via devtools or a screen recorder.

4 Uploading video files

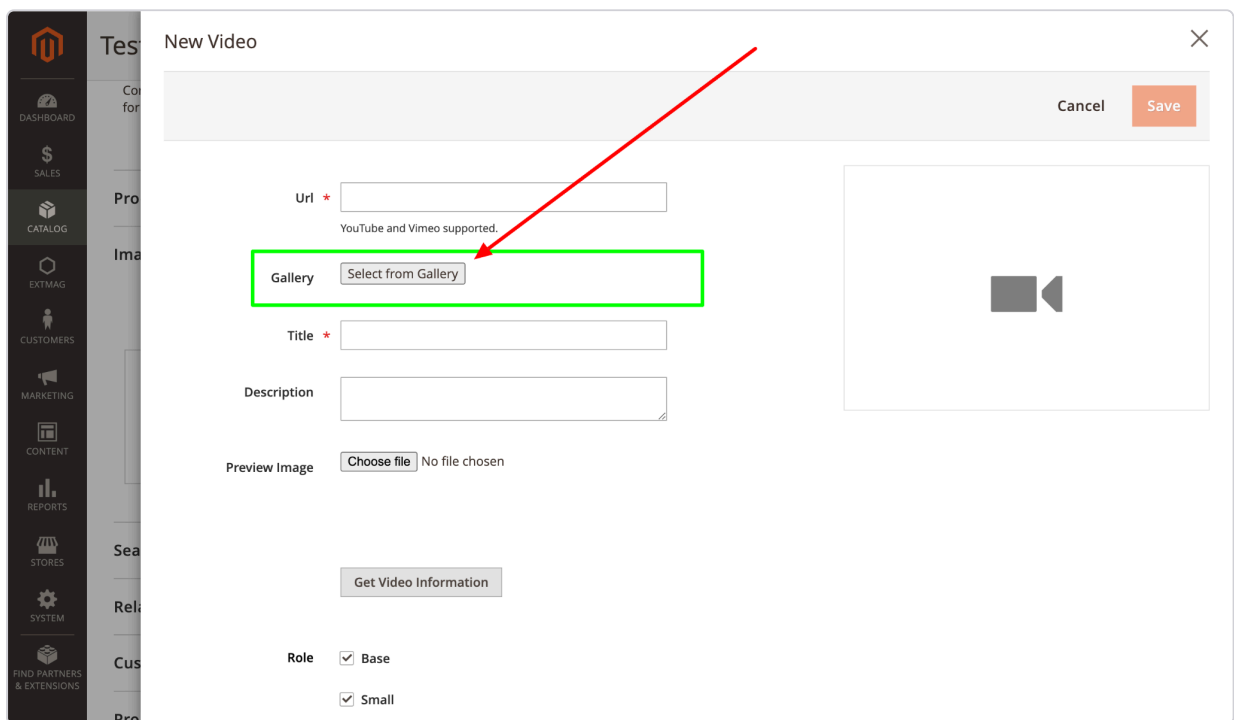
There are two entry points for uploading a video. Both lead to the same WYSIWYG file browser dialog (`Insert File...`), so a video uploaded from one is immediately available to the other.

4.1 From a product edit page

1. Open a product → expand the **Images And Videos** fieldset.
2. Click **Add Video** (top right of the fieldset).
3. In the *New Video* dialog click **Select from Gallery**.



Product Images And Videos — Add Video button

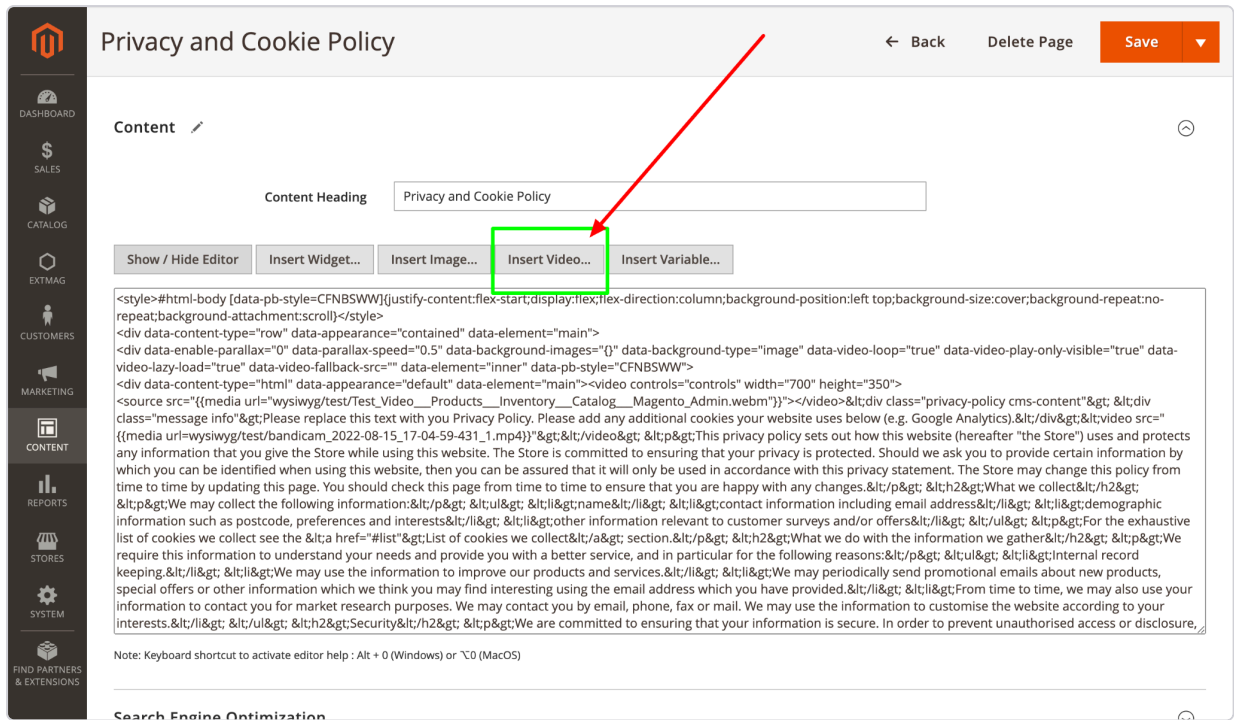


New Video dialog — Select from Gallery

4.2 From a CMS page, block or Page Builder content

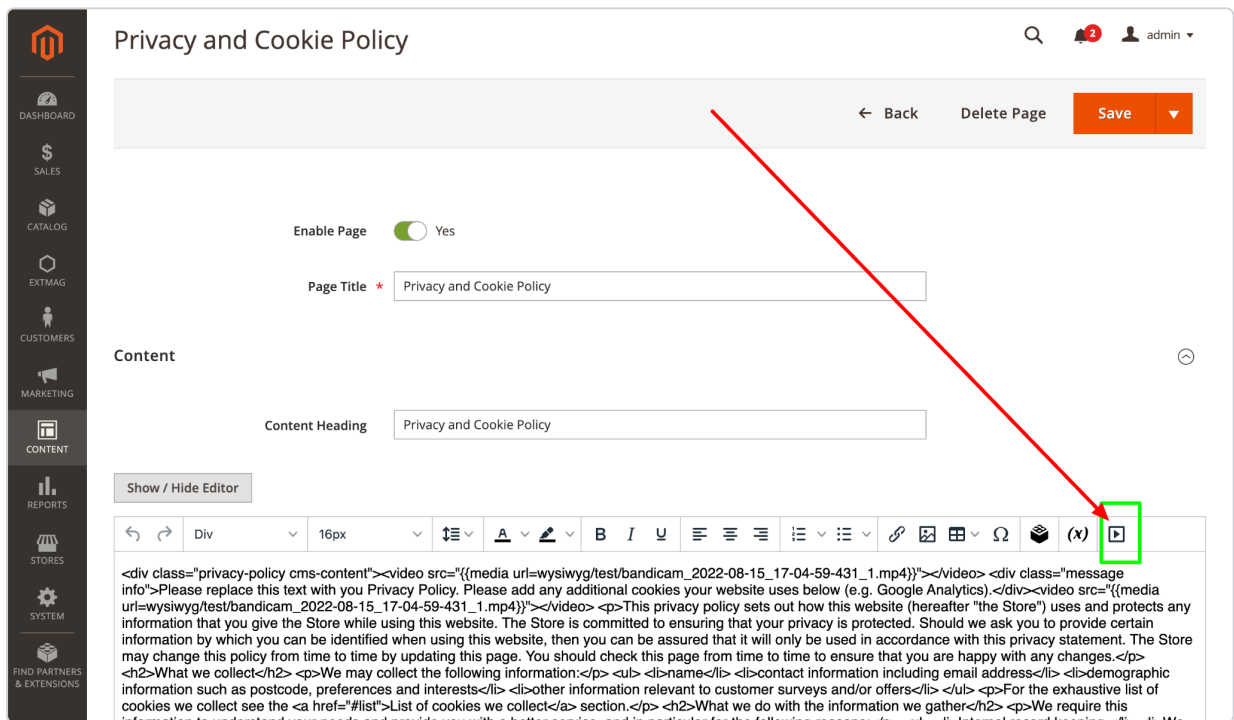
1. Open the CMS page / block edit form.
2. Place the cursor where the video should appear.
3. Click **Insert Video** in the WYSIWYG toolbar.

In the standard CMS / Block editor the button sits in the top toolbar:



Insert Video button — CMS toolbar (top toolbar position)

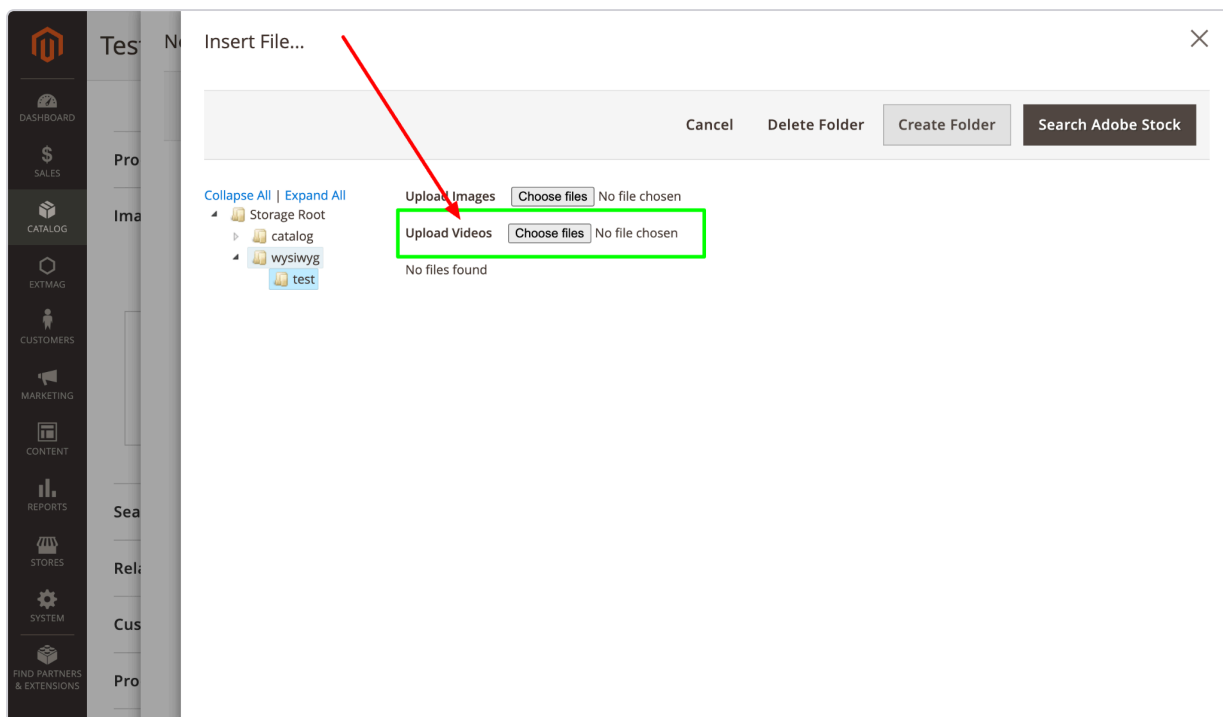
In the TinyMCE 5 floating toolbar (some Page Builder contexts) the button sits in the secondary row:



Insert Video button — TinyMCE 5 secondary row

4.3 File browser — Upload Videos

In both paths the *Insert File...* dialog opens. The module adds an **Upload Videos** row alongside *Upload Images*:

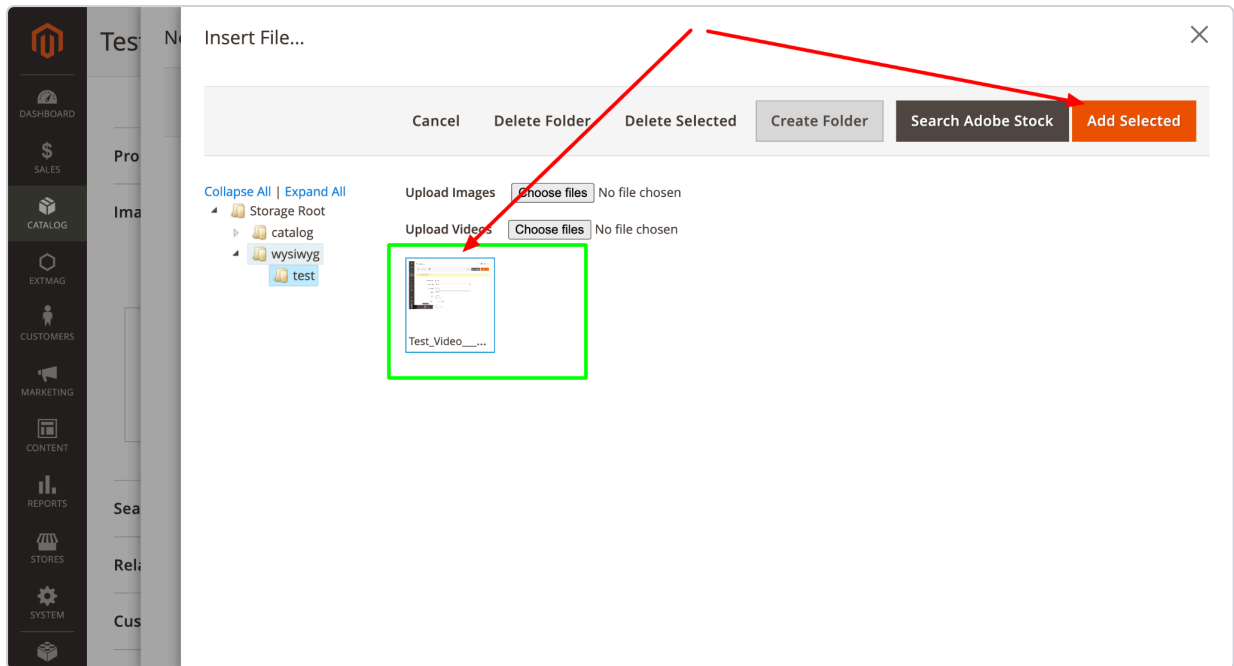


Insert File dialog — Upload Videos row

- Click **Choose files** next to *Upload Videos* and pick the local file.
- The upload uses the standard Magento file uploader; large files need `upload_max_filesize` / `post_max_size` raised in `php.ini` (see the *Installation Guide*).
- The file is written under `pub/media/wysiwyg/<folder>/` — pick or create a subfolder from the left tree before uploading to keep things organised.

5 Selecting a previously uploaded video

After upload (or when reusing an existing video) the file appears in the folder grid. Click the thumbnail and then **Add Selected**:



Insert File dialog — Add Selected after picking a video

For the **product gallery** path: the chosen file becomes the video source on the *New Video* form. Fill in *Title* (required), optional *Description* and *Preview Image*, pick the gallery *Role* (Base / Small / Thumbnail / Swatch) and click **Save**. The video is now a media gallery item for the product.

For the **WYSIWYG** path: a `<video>` tag is inserted at the cursor position with `src="{{media url=...}}"`. The editor renders it as a visible block so you can move / delete it like any other element.

6 Editing the embedded `<video>` tag

The default embed is a bare `<video>` element:

```
<video src="{{media url="wysiwyg/folder/file.mp4"}}"></video>
```

You can add any HTML5 video attribute by hand. Common ones:

Attribute	Effect
<code>controls</code>	Show the browser's default playback controls.
<code>autoplay</code>	Start playback when the page loads. Most browsers require <code>muted</code> too.
<code>muted</code>	Start muted (required for <code>autoplay</code> on most browsers).
<code>loop</code>	Restart playback when the file ends.

Attribute	Effect
<code>playsinline</code>	Play inline on iOS Safari instead of going fullscreen.
<code>preload</code>	<code>none</code> / <code>metadata</code> / <code>auto</code> . Defaults to browser preference.
<code>poster</code>	URL of an image to show before playback starts.

Worked example — an autoplaying, muted, looping background video with no controls:

```
<video
  src='{{media url="wysiwyg/folder/bg.mp4"}}'
  autoplay muted loop playsinline></video>
```

A worked example — a controllable player with custom poster and a fallback:

```
<video controls preload="metadata"
  poster="{{media url='wysiwyg/folder/cover.jpg'}}">
  <source src="{{media url='wysiwyg/folder/promo.webm'}}" type="video/webm">
  <source src="{{media url='wysiwyg/folder/promo.mp4'}}" type="video/mp4">
</video>
```

The *Controls List* tokens are added automatically at render time — you do not need to repeat them in the `<video>` tag.

Full HTML5 `<video>` attribute reference: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/video>.

7 YouTube and Vimeo embedding

The native Magento product video flow accepts a YouTube or Vimeo URL in the *Url* field of the *New Video* form. The module improves the playback experience (cleaner iframe, fewer layout jumps on mobile, Hyva compatibility) but does not change the workflow:

1. Open the product → *Images And Videos* → *Add Video*.
2. Paste the YouTube / Vimeo URL into **Url**.
3. Click **Get Video Information** — Magento fetches the title / description / preview.
4. Save.

For CMS content, paste the iframe embed from YouTube / Vimeo directly into the WYSIWYG source view. The *Insert Video* button is only for self-hosted files.

8 Storefront behavior

Self-hosted videos appear in the product gallery alongside images. The slot order follows the order shown in the *Images And Videos* fieldset — drag to reorder.

When the *Controls List* configuration is non-empty, the storefront JavaScript (`Extmag_Videopro/js/apply-controlslist`) walks every `<video controls>` element on page load and applies the configured attributes. Videos added by AJAX (quick view, gallery swap) are picked up by a MutationObserver.

9 Troubleshooting

9.1 Upload Videos row missing

- Confirm the module is enabled: `bin/magento module:status Extmag_Videopro`.
- On Magento 2.4.7+, run `bin/magento setup:static-content:deploy -f` — the upload row is rendered from a static template override.
- Hard-refresh the admin browser (Cmd+Shift+R / Ctrl+Shift+F5) so cached JS templates reload.

9.2 Uploaded file rejected with *Disallowed file type*

The extension is not in *Allowed Video Formats*. Add it under `Stores → Configuration → Extmag → Video Pro → General` and save.

9.3 Video uploaded but does not play on the storefront

- Inspect the `<video>` tag in browser devtools. The `src` URL must resolve (200, correct `Content-Type`).
- Try MP4 (H.264 / AAC). Most cross-browser issues come from formats with no native HTML5 support (AVI, WMV, FLV) or unusual codecs in an MP4 container.
- Empty *Allowed Video Formats* does not block playback of already embedded videos, only further uploads. If a working video stops playing after upgrade, check the storefront error log and server-side `Content-Type` headers.

9.4 Picture-In-Picture still appears after enabling *Disable Picture In Picture*

- `bin/magento cache:flush` and a browser hard refresh — the storefront config block is cached.
- Firefox enforces `disablePictureInPicture` partially; some builds ignore the hint entirely. There is no cross-browser way to fully remove the PiP entry point.

9.5 Large uploads fail silently or with HTTP 413

Raise `upload_max_filesize`, `post_max_size` (PHP) and `client_max_body_size` (nginx). Restart PHP-FPM and the web server. See the *Installation Guide* for details.

9.6 *Insert Video* button missing from TinyMCE

- Run `bin/magento cache:clean config block_html layout` and reload the CMS editor.
- The button is only registered for the Magento-bundled TinyMCE 5 editor. Third-party WYSIWYG editors are not supported.

9.7 Hyva theme — gallery thumbnail shows but player does not load

Install the Hyva-themes compatibility adapter for `Extmag_Videopro`. The base module wires the Luma `fotorama` gallery; Hyva renders media through its own Alpine.js component and needs a small bridge.