

The Arc Embed — Technical Specification

The player URL, the iframe attributes, streaming, captions, the security model and its limits, the data captured, payments, and what a client website must allow.

Version 1.0

October 2026

For developers and technical reviewers

Contents

1. Scope and versioning
2. How it works
3. Player URL and parameters
4. iframe attributes
5. Player features
6. Streaming and delivery
7. Captions and subtitles
8. Supported browsers
9. Host page events (optional)
10. Security model and limitations
11. Data captured
12. Payments
13. Client site requirements
14. Support and change log

1 Scope and versioning

This document specifies version 1.0 of The Arc Embed, the PitoReel embeddable player, released October 2026. The stable interface for a client website is the player URL (section 3) and the iframe attributes (section 4).

The player itself runs on PitoReel. Improvements and fixes reach every embed automatically, with no change to a client's page. A change to the URL or the required attributes would be published as a new version of this document and announced to clients in advance.

2 How it works

Your web page

Contains one iframe pointing at the PitoReel player URL, with your client key and a title id.



The Arc Embed player

Served by pitoreel.com. Checks the key, the website it is on and the title, then asks for a short-lived playback token.



PitoReel streaming

Adaptive video delivered over the Amazon CloudFront content delivery network.

1. The visitor's browser loads the iframe from `https://pitoreel.com/embed`.
2. PitoReel records which website the iframe is on, and the browser is told which websites may show it for this key.
3. The player requests its configuration. PitoReel checks that the key is live, that the website is approved and that the title is allowed, and replies with the poster, the title and what may play.
4. When playback starts, the player obtains a signed playback token valid for five minutes and renews it automatically while the visitor watches.

3 Player URL and parameters

Player URL

```
https://pitoreel.com/embed?key=YOUR_CLIENT_KEY&id=TITLE_ID
```

Query parameters

Parameter	Required	Set by	Meaning
<code>key</code>	Yes	Client, from PitoReel	The client key. It decides the approved domains, the titles, the access mode and the branding.
<code>id</code>	Yes	Client, from PitoReel	The id of the title to show.
<code>access</code>	No	PitoReel only	Carried by the link in a pay-per-view buyer's receipt email. Never add it to embed code.

The player reads no other parameters. Appearance and playback options are part of the key, set by PitoReel, not part of the URL. If `key` or `id` is missing the player shows "This player is not configured".

4 iframe attributes

Fixed size — 640 × 360

```
<iframe src="https://pitoreel.com/embed?key=YOUR_CLIENT_KEY&id=TITLE_ID"
  width="640" height="360"
  frameborder="0"
  allow="autoplay; fullscreen; picture-in-picture"
  allowfullscreen
  title="PitoReel player"></iframe>
```

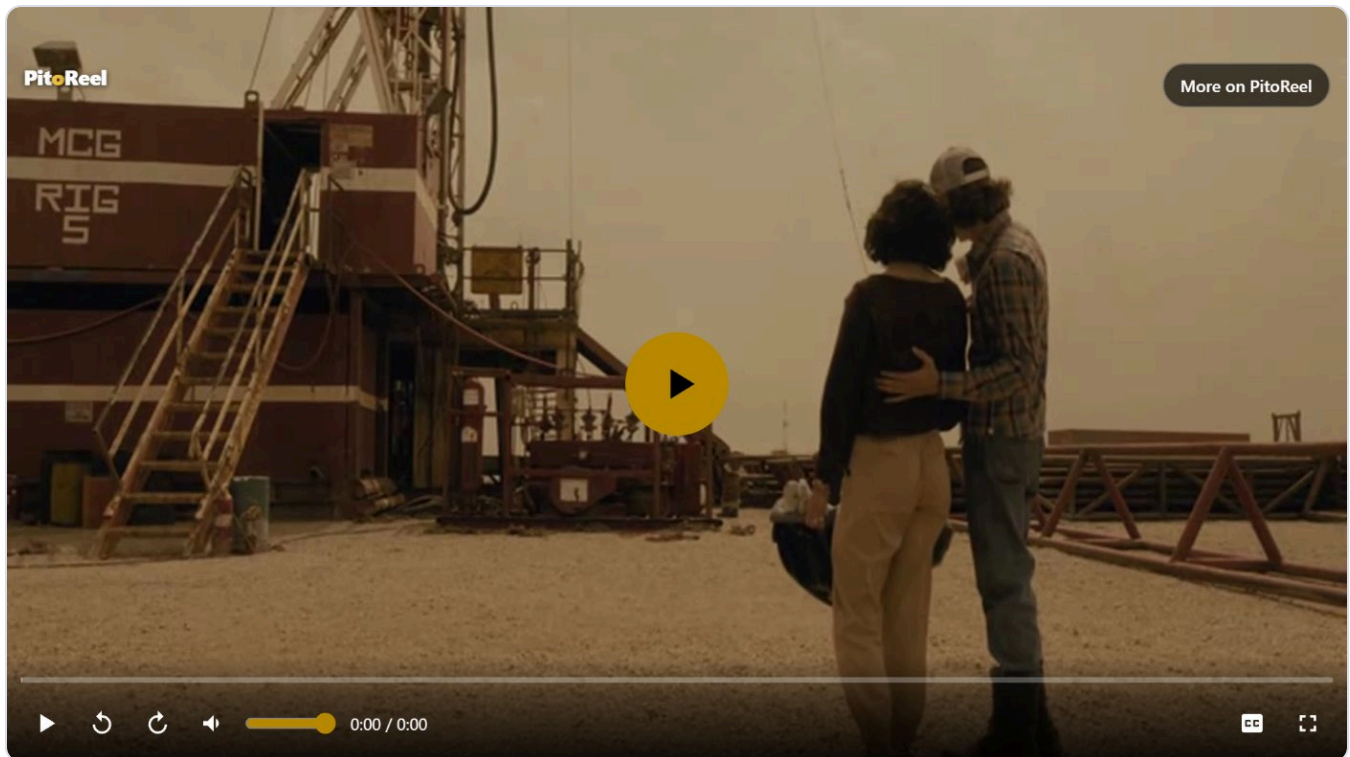
Responsive — fills its container at 16:9

```
<div style="position:relative; width:100%; padding-top:56.25%;">
  <iframe src="https://pitoreel.com/embed?key=YOUR_CLIENT_KEY&id=TITLE_ID"
    style="position:absolute; top:0; left:0; width:100%; height:100%; border:0;"
    frameborder="0"
    allow="autoplay; fullscreen; picture-in-picture"
    allowfullscreen
    title="PitoReel player"></iframe>
</div>
```

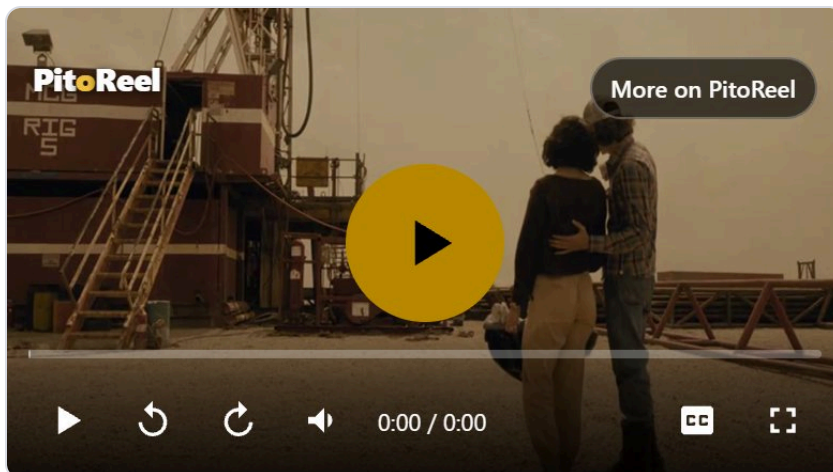
Attributes

Attribute	Value	Why
<code>src</code>	The player URL (section 3)	Loads the player.
<code>width</code> , <code>height</code>	<code>640</code> , <code>360</code> , or the responsive wrapper	The player is 16:9. Any size keeps that shape; the picture letterboxes if the box differs.
<code>allow</code>	<code>autoplay</code> ; <code>fullscreen</code> ; <code>picture-in-picture</code>	Required. Lets the player start, go full screen and use picture-in-picture.
<code>allowfullscreen</code>	(boolean)	Required for full screen in browsers that read the older attribute.
<code>frameborder</code>	<code>0</code>	No border around the player.
<code>title</code>	A short label	Announced by screen readers. PitoReel fills it with your key's label; you may use the film's name.
<code>sandbox</code>	Do not add	The player needs scripts, a payment window and full screen. A sandboxed player will not work.

5 Player features



The player at desktop width, before playback.



The same player at 390 pixels wide. One responsive player; no separate mobile build.

- No site chrome — only the player. The title's poster is shown until playback starts.
- Controls: scrub bar with buffered progress, play and pause, back 10 seconds, forward 10 seconds, mute, volume slider (hidden on narrow screens, where the device's own volume buttons are used), elapsed and total time, a subtitles menu when the title has tracks, a quality menu when it has more than one rendition, and full screen.

- The control bar hides itself about three seconds into playback and returns on any movement, tap or key press. It is always shown while paused.
- A small PitoReel mark at top left and a “More on PitoReel” link at top right, both kept on the picture even when a film letterboxes. Either can be turned off on the key.
- Autoplay, when enabled on the key, starts muted, because browsers block autoplay with sound.

Keyboard

Key	Action
Space or K	Play / pause
M	Mute / unmute
F	Full screen
Left / Right	Back / forward 10 seconds (5 seconds when the scrub bar has focus)

What the player shows instead of a blank frame

Situation	The player shows
The website is not approved for the key	“Not authorised on this website”, naming the domain it saw
The key has passed its end date	“This embed has expired” — “Please contact PitoReel to renew it.”
The key has been switched off, or does not exist	“Cannot play this title”, with the reason underneath
The code is missing its key or title id	“This player is not configured”
A trailer key on a title with no trailer	“No trailer available”
The title is not ready to play yet	“Not available yet”
The network failed while starting	“Could not reach PitoReel”

6 Streaming and delivery

- **Adaptive HLS** streaming, played with hls.js or the browser's native HLS support (Safari); progressive MP4 where a title has no HLS rendition.
- **Renditions** at 480p, 720p and 1080p where available. The player chooses automatically and offers a quality menu when more than one exists.
- **16:9** player. Titles in other shapes letterbox inside it; nothing is stretched.
- **Delivery** over the Amazon CloudFront content delivery network. Your web server is not in the video path and needs no extra bandwidth, however many people watch.

7 Captions and subtitles

- Speech captions, and translated subtitles in **Spanish, French and German**, where the title has them. Not every title has them.
- Machine-generated tracks carry the label **(Auto)** after the language name in the subtitles menu.
- The subtitles menu appears only when the title has at least one track.

8 Supported browsers

Current versions of the major evergreen browsers — Chrome, Edge, Firefox and Safari — on desktop and on mobile phones and tablets. JavaScript must be enabled.

9 Host page events (optional)

A page that wants to react to the player, or control it, can use `window.postMessage`. No code is needed for ordinary use, and no viewer data is ever sent in these messages.

Commands your page can send to the player

Message	Effect
<code>{ action: 'play' }</code>	Start playback
<code>{ action: 'pause' }</code>	Pause
<code>{ action: 'seek', seconds: 30 }</code>	Jump to a position, in seconds
<code>{ action: 'mute' }</code> / <code>{ action: 'unmute' }</code>	Sound off / on
<code>{ action: 'volume', value: 0.5 }</code>	Volume from 0 to 1

Events the player posts to your page

event	Extra fields
<code>ready</code>	<code>title</code> , <code>asset</code> , <code>duration</code>
<code>play</code> , <code>pause</code>	<code>time</code>
<code>timeupdate</code>	<code>time</code> , <code>duration</code>
<code>ended</code>	—
<code>error</code>	<code>message</code>
<code>paywall</code>	<code>title</code>
<code>purchase_started</code>	<code>contentId</code>
<code>adstart</code> , <code>adend</code>	<code>advertiser</code> (on <code>adstart</code>)

Example

```
var player = document.querySelector('iframe[src^="https://pitoreel.com/embed"]');

// Control the player.
player.contentWindow.postMessage({ action: 'play' }, 'https://pitoreel.com');

// Listen to it. Every message from the player carries pitoreel: true.
window.addEventListener('message', function (e) {
  if (e.origin !== 'https://pitoreel.com' || !e.data || !e.data.pitoreel) return;
  if (e.data.event === 'ended') { /* for example, show your next call to action */ }
});
```

Events describe playback only. The player also supports sponsor placements arranged by PitoReel; none run today, so `adstart` and `adend` are not currently sent.

10 Security model and limitations

What protects the content

- **Per-key domain allowlist.** When the iframe loads, PitoReel reads which page it is on, signs that record, and checks it against the key's approved domains on every configuration and playback request. A tampered record is refused outright. Subdomains of an approved domain are included; a look-alike such as `notexample.com` never matches `example.com`.
- **Browser-enforced framing.** The player's response carries a `Content-Security-Policy: frame-ancestors` header listing the key's domains, so the visitor's browser itself refuses to show the player anywhere else.
- **Short-lived playback tokens.** Every playback needs a signed token minted by PitoReel, valid for five minutes and renewed automatically while someone watches. A copied video address stops working within minutes. The key alone grants nothing.
- **Server-side checks** on every configuration, token, play and progress request.
- **Key controls.** A key with no approved domain plays nowhere; pay-per-view and licensed keys must list their exact titles; a key can carry an end date and can be switched off instantly.

Limitations, stated plainly

The protection stops link sharing, hotlinking and use of a key on a website it was not issued for.

It does **not** prevent screen recording. There is **no DRM** in version 1.0.

11 Data captured

What is recorded

Record	Contents	Personal data
Per-key counters	Player loads, plays started, purchases, last-seen date and domain	None
Playback beacons	Once per playback, plus progress in 5% steps: key, title, an anonymous session id, the embedding domain, and whether it was the trailer or the full title	No viewer name or account
Pay-per-view purchase	Buyer email (for the receipt and the return link), amount, currency, the Stripe payment id, the domain it was bought on, watch time; IP address and browser for fraud review only, never in analytics	Buyer email

- The secret in a buyer's return link is stored only as a SHA-256 hash; the link itself is emailed once.
- A purchase is linked to a PitoReel account only if the buyer happens to be signed in to PitoReel.
- Reports on plays, sales and revenue per key are provided by PitoReel on request. There is no client login.
- Consumer privacy policy: pitoreel.com/privacy.html.

12 Payments

Pay-per-view is available by arrangement and enabled by PitoReel per key.

- Card payment through **Stripe**. Card details are entered in a field served by Stripe and never reach PitoReel's servers or the client's, which keeps PitoReel in the lightest PCI scope (SAQ-A).
- The price is set on the key, server-side; the browser can never set it.
- Access is granted only after PitoReel confirms with Stripe directly that the payment succeeded.
- Checkout runs in a PitoReel window opened by the player, not inside the client's iframe, so card authentication challenges work on any website. If the browser blocks the window, the player offers to open the checkout in a new tab.
- No account is needed. The buyer receives an email link that is the receipt and the way back to the film. Access is permanent by default; PitoReel can set a 48-hour, 7-day or 30-day window per key.

13 Client site requirements

Checklist

Requirement	Detail
HTTPS	Serve the page over HTTPS.
Approved domain	The page must be on a domain approved for the key. Test on that domain.
Content-Security-Policy	If the site sends one, its <code>frame-src</code> list must include <code>https://pitoreel.com</code> .
Attributes	Keep <code>allow</code> and <code>allowfullscreen</code> ; never add <code>sandbox</code> .
Container	Give the player a 16:9 space, or use the responsive wrapper.
Site builders	Paste into an HTML, Custom HTML or Code block. A builder that wraps custom HTML in its own frame on another domain may be reported as that domain — contact PitoReel.

Content-Security-Policy, if your site sends one

Content-Security-Policy: `frame-src https://pitoreel.com`

14 · Support and change log

Email support@pitoreel.com. Online: pitoreel.com/player.html

1.0 · October 2026 · First release.

The Arc Embed — Technical Specification · Version 1.0 · October 2026 · © 2026 PitoReel. All rights reserved.
Installation steps and troubleshooting are in the companion document, The Arc Embed — Integration Guide.