

Data classification and handling

Version	1.0
Effective date	27 October 2026
Owner	Robin Ashford
Applies to	All data Final Frame stores, processes or sends
Review	27 October 2027
MPA CSBP	OR-1.4, OP-3.0, OP-3.1, TS-1.0, TS-1.16, TS-3.0

Approved by Robin Ashford, Founder, 27 October 2026

Classes

Class	What it covers	Examples
C1 Content (highest)	Customers' media and anything derived from it	Masters, mezzanines, proxies, audio, subtitles, dubs, trailers, artwork, stills, frames, film reading copies, generated candidates made from them
C2 Restricted	Commercial and personal data that is not media, and anything that grants access	Deal memos and terms, rights, project inbox emails and attachments, the audit trail, user names and emails, Screening Room guests and viewing records, secrets, keys, session tokens, signed URLs, API keys, database backups
C4 Public	Published on purpose	Marketing site, public docs, public catalogue data from Wikidata, TMDB and others

When in doubt, use the higher class. A file that mixes classes takes the highest.

Handling rules

Rule	C1 Content	C2 Restricted	C3 Internal
Where it is stored	The Org's own bucket only. The scratch bucket only for a reading copy, deleted within 1 day	Postgres with RLS, Secret Manager, the Org's bucket	GitHub, Cloud Logging
Encryption at rest	Per-Org CMEK (Cloud KMS, AES-256).	Google-managed AES-256 (Cloud SQL, Secret Manager). Production Cloud SQL: CMEK	Provider default
Encryption in transit	TLS 1.2 or higher	Same	Same
How it moves	Signed URLs only, one object, short lived (15 minutes default, 60 maximum). Never through the API	Through the API, after authorise()	
Who can see it	Members of the Org with asset.view or asset.download, scoped by Group and Project	Members with the matching permission. Secrets: the service and Robin	Robin
Sent to a third party	Only to vendors approved for C1 in the Vendor management policy and the AI use policy	Only to approved Tier 1 or 2 vendors	Approved vendors
On a workstation	Not by default. Only for a task that needs it; deleted after	Only what the task needs	Allowed

Rule	C1 Content	C2 Restricted	C3 Internal
In logs	Never	Never secrets, tokens, cookies, signed URLs or full bodies. User email on failed sign-in only	Allowed
In tests or local dev	Never. Seed data only	Never real values	Allowed
Deleted	See the Data retention and disposal policy	Same	Same

Deal memos being read

A memo is read by the memo.read job. Before a deal exists, the New project form's memo is held with its reading so the job can read it, and so the deal created afterwards can take it. C2 Restricted, like the memo itself.

Data	What it holds	Handling
Memo readings	The draft memo's bytes and their hash, the file's name and type, the reading's progress, each part's proposal, the merged proposal, which model, region and provider answered, who started it, and the request	Org-scoped, RLS forced. A draft's reading is its reader's alone. The bytes go when a deal takes the reading or it is discarded; every row goes a day after it was made (the worker's sweep). Never in a log, a job payload or a response. The house memo proof reads memos where they live and prints scores, never content

Screening Room guests and viewing records

The Screening Room holds personal data about people outside the customer: the buyers it invites as screening guests. It also records what every viewer, member or guest, opens and plays. All of it is C2 Restricted. Screeners and trailers stay C1 Content in the Org's bucket.

Data	What it holds	Handling
Guest identity	Email address, name, company, expiry, whether the NDA is asked and the version accepted with its time, who invited them and when, the last invitation, first sign-in, last seen, status, revocation with who and the reason, the lock counters	Org-scoped, RLS forced. Read in the CMS by screening.guests.manage or screening.manage. Never joined to users
Guest credentials	Link and session tokens as SHA-256 hashes, the asking browser's binding as a hash, the IP that asked for a link, passkey credential ids, public keys and counters, authenticator secrets sealed with, session IP and user agent	The database alone can open none of it. No token, secret or code is returned after enrolment or written to a log
Viewing records	Each guest sign-in, sign-out and NDA acceptance, and each title view, play start, progress and finish, and search result opened by a member or a guest: when, who, the viewer session, the play, the title, position and duration, audio and subtitle language, IP address, user agent, request id, and a small detail (the sign-in method, the deliverable, the words of a search)	Append only. Read by the CMS's guest Activity and Analytics only; no viewer route and nothing on the viewer host reads it. In Analytics a person is named only to screening.manage, a guest's company only to screening.guests.manage
Viewers' own lists	My list, recent searches	Each viewer's own. The viewer clears their searches
The plays export	The Analytics plays table as CSV, at most 10,000 rows	Same privacy as the screen. Cache-Control: no-store.

- **Field policy.** Each audience's hidden fields are removed by the server before an answer leaves. By default guests never see the budget, internal notes or rights holders.
- **Email.** A guest's address goes to Resend for the invitation and for each sign-in link. The invitation carries no token. The sign-in link's token sits in the address's fragment.
- **Models.** No guest data, viewing record or screener goes to a model. The theme pass sends catalogue facets and counts only (see the AI use policy).
- **Logs.** The door's and the viewer's log lines name a guest by id and an internal user by actor id. Never a token, cookie, code or signed link.
- **In the browser.** The viewer host keeps nothing in browser storage. Each Org has its own host-only cookies.

Customer cloud credentials

A customer may connect its own S3, Google Cloud Storage or Azure buckets. The credential they give us is C2 Restricted and is handled as follows.

- **At rest:** AES-256-GCM, a fresh data key per credential, the data key wrapped by the platform key in Cloud KMS (europe-west2, rotated every 90 days). The database holds the wrapped key and the ciphertext. KMS holds the key. Neither alone reads a credential.
- **In use:** opened inside the provider call, held for that call, never returned by a route, never written to a log or an audit row. Audit rows record provider, bucket, name and the shape of the credential.
- **Least privilege, what we ask the customer for:** a read-only identity scoped to one bucket and, where possible, one prefix. On Google: a service account with Storage Object Viewer on the bucket, never a project owner or editor key. On AWS: an IAM user or role with s3:ListBucket and s3:GetObject on that bucket alone, never root keys. On Azure: a container SAS with read and list rights and an expiry, or a storage account key for that account alone. The platform's own identity is used for the demo library only and is refused for any customer bucket of ours.
- **Rotation:** a replaced credential re-encrypts and marks the connection untested until the Test button passes. The platform key rotates automatically; old versions stay to unwrap what they wrapped.
- **Revoke:** disable the connection (nothing browses, imports or watches through it), then revoke the key at the provider. Disabling is immediate; the row stays so every import keeps its connection.
- **Where it travels:** to the provider's own endpoint over TLS and nowhere else. No third party sees a customer credential.

Labelling

The platform labels nothing in the file. The class follows from where the data lives: anything in an Org bucket is C1. Exports and reports leaving the platform carry the Org name and "Confidential".

Residency

Content and database: europe-west2 (London). Processing elsewhere is listed, with its location, in the AI use policy or the Vendor management policy: model calls on Vertex AI's global endpoint and in europe-west4, fal.ai for GPT Image, and Cloud Build staging in the US (source code only).