

Security facts, with the boundaries visible

CREDENSHARE • DOCUMENTATION SNAPSHOT • 29 SEPTEMBER 2026

Use this briefing to identify the product path and the questions relevant to your evaluation. It summarizes the official pages cited below; it is not an independent audit, penetration test or inspection of a customer's installation. Confirm requirements against current product behavior before relying on them.

Where encryption happens

Creation path	Documented processing	Evaluation consequence
Web Share	The browser encrypts fields with AES-256-GCM before upload.	Protect the complete reading link and the browser environment.
REST API / official SDK Share	Client encryption precedes the API request; plaintext creation is refused.	Use the documented client format or supported SDK, not a plaintext shortcut.
Slack-native Share	Content passes through Slack and is encrypted on CredenShare's server.	This is not the same trust boundary as browser/client creation.

The encryption documentation says the client content key is carried in the URL fragment and the service stores ciphertext plus an access-token hash. A standard browser request does not transmit that fragment, but a person or script that can read the complete link can expose it. [C1]

Two different flows

Sending a Share: approved sender holds the information client encrypts service stores/serves ciphertext authorized recipient's browser uses the complete link to decrypt. The operator must protect the reading link.

Collecting through Secure Requests: requester creates a form/keypair submitter uses the Collect link submitter's browser encrypts to the request public key only the account that created the request can open the submissions, using the request's private key (cached in the browser that created the request, recovered through zero-knowledge custody, or carried in the saved Access link). The Collect link carries no key and is the one to send; the Access link is for the creator to keep. A request created through the MCP tool **request_secret_from** has no keypair or Access link: its submissions are encrypted on CredenShare's servers, so CredenShare can read them (MCP server). The documentation places Secure Requests on Business and Enterprise. [C2]

Decision: Choose the creation path before accepting a security statement about it. Do not generalize a browser/API property to Slack-native shares or to requests created through MCP.

[C1] [Encryption](#). [C2] [Secure Requests](#). Flow descriptions are explanatory summaries, not infrastructure diagrams verified against production.

Access, continuity and evidence

Encryption is not the same as custody

Client encryption protects content before upload. Optional **zero-knowledge custody** is a separate continuity mechanism: wrapped keys allow approved activated devices to reconstruct access. CredenShare stores those wrapped keys, including the account key sealed under the customer's passphrase, but does not receive the passphrase that opens it. The current custody guide describes Business/Enterprise eligibility, non-retroactive coverage and no support recovery of a lost passphrase. [C3]

For an operational dependency, demonstrate the actual second-device and backup-operator behavior with disposable data. Do not assume a team role guarantees the ability to read another person's content. Establish who stores recovery material and which actions are destructive.

Controls have specific scope

Requirement to check	What the cited documentation establishes
Inactivity controls	The team owner can set an inactivity limit. This is not a statement that all account protections are mandatory.
IP restriction	It is a per-share control on entitled plans, not an organization-wide allowlist applied to all shares.
Access history	It includes successful opens and wrong-passcode attempts. An IP refusal happens before that logging path.
Export coverage	Exports are creator/caller-scoped and subject to the applicable readable window and per-export cap.

The policy/audit guide distinguishes a **readable log window** from deletion of historical rows. A plan window must not be described as a promise that records are erased at its end. An anonymous open is not verified business authorization, and an empty log does not prove no one attempted access. [C4]

Six questions for your evaluation

1. Which creation and retrieval path will this workflow actually use?
2. Can the intended recipient satisfy its required protections?
3. Which continuity route has been rehearsed successfully?
4. Whose logs can be read, over what window, with what omissions?
5. What separate system ends the underlying credential's access?
6. Which contractual or vendor-assurance requirement remains unfulfilled?

[C3] [Zero-Knowledge Custody](#). [C4] [Policy and Audit](#).

Retention and vendor-assurance boundaries

Do not equate payload expiry with erasure everywhere

The encryption page describes hourly live-ciphertext cleanup and seven-day production database backups. The compliance page describes retained metadata after expiry. The policy/audit page separately describes access-log retention. These are different data classes. Expiry does not erase a recipient's copy or revoke access at the system that issued a credential. [C1, C4, C5]

Current disclosed operating and assurance limits

As of the cited documentation snapshot, CredenShare describes US East (N. Virginia) hosting, single-AZ production database operation, no alternate-region setting and no published uptime measurement/SLA. It discloses no SOC 2 report, ISO certificate, HIPAA BAA or independent penetration-test report. It also reports outstanding administrative-access hardening and account-level API-audit work. These are disclosed limits, not controls this briefing claims have been completed. [C5]

A required independent report or residency condition must be evaluated directly. A resource pack, encryption description or sample evidence packet cannot substitute for a missing contractual or assurance requirement.

Record a scoped conclusion

For [workflow], we evaluated [creation path] against [requirements]. Demonstrated checks: [safe references]. Documentation relied on: [version/date]. Unresolved requirements: [specific gap and owner]. Decision: [approved scope, limited pilot, or not approved].

Use that structure after your own evaluation; this blank wording asserts no result. Do not send credentials or live reading links with a security question.

Source register

Code	Official page
C1	Encryption
C2	Secure Requests
C3	Zero-Knowledge Custody
C4	Policy and Audit
C5	Compliance and current limitations

Review discipline: recheck these pages and the chosen product path when the requirement or implementation changes. This document is a dated briefing, not a guarantee that later releases behave identically.