

ENGINEERING PLAYBOOK

Encrypted Cloud Storage Explained

Layered encryption without performance compromise: what two encryption layers cover, why Everabyte's storage infrastructure holds Vault-backed master keys, and how owner-controlled decryption and recovery passphrases keep you in charge.



This paper describes Everabyte's encryption model as it is built today, without claiming any certifications. It contains no invented statistics - every product statement is from Everabyte's own engineering documentation.

CONTENTS

In this paper

01	What Layered Encryption Covers	3
02	Client-Side vs Layered Models	3
03	Everabyte Encryption Architecture	5
04	Performance Considerations	6
05	Storage Regions and Residency	6
06	Compliance Mapping (GDPR)	7
07	Implementation Guide	8
08	Free Audit Checklist	8
	About this document	9

How to read this paper

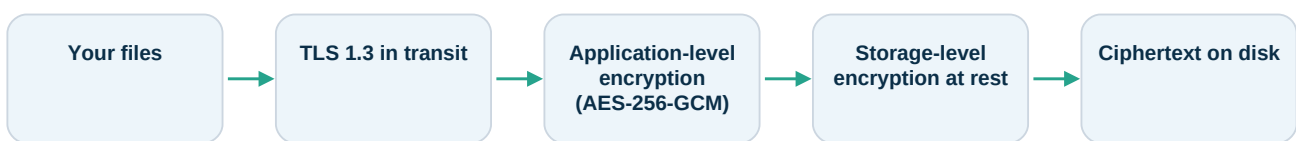
Chapters 1-3 explain what Everabyte's layered encryption covers and the products it does not include; Chapter 4 stays qualitative and promises no invented speed figures; Chapters 5-8 cover residency, GDPR control mapping, an implementation guide and an audit checklist you can run before purchase. This is security-engineering documentation, not legal advice.

What Layered Encryption Covers

Two layers of encryption protect your stored files - one at the application level and another within Everabyte's storage infrastructure. Your decryption key remains under your control: Everabyte services cannot use it or access decrypted stored files without an operation authorized by you.

That single sentence is the model, and this paper unpacks it honestly: what each layer does, what the key management structure looks like, where the two-layer approach differs from client-side and zero-knowledge models, and what to ask when a security review probes the model.

On the path into Everabyte's storage infrastructure, each layer fills a different role:



- In transit: connections are protected with TLS 1.3 - the same connection-level protection that carries login sessions.
- Application level: Everabyte's application services apply AES-256-GCM encryption to your files before they are written to storage.
- Storage level: an additional encryption-at-rest layer is applied within Everabyte's storage infrastructure to files already encrypted at the application level.
- The result: direct access to stored objects does not reveal the original file contents.

Defense in depth, stated plainly

The two layers are independent by construction: a compromise that removed one encryption context would still leave a second barrier between stored ciphertext and readable data. This is a layered design rather than a zero-knowledge claim - see Chapter 2 for the full model comparison.

Client-Side vs Layered Models

"Encrypted cloud storage" is decided by one question: when does encryption happen, and who holds the keys? Four distinct models answer that question differently - and they are not interchangeable. The table below is the comparison that most customer security reviews turn on.

Model	Encryption happens	Keys	Considerations
Zero-knowledge (client-side)	On the end-user's device, before upload	Only the user holds them	Platform cannot process content without the user's key; lost keys or passphrases can mean lost data
Client-side E2EE	At the edges of the protected communication (sender to receiver)	End-user holds them	Strong against server compromise; workflow features (search, processing) must be reimplemented around ciphertext
Server-side layered (Everabyte)	Server-side: once by the application before storage, once at rest within the storage infrastructure	Application decryption keys you control; Vault-managed master keys for Everabyte's infrastructure	Platform can process authorized operations; direct access to stored objects does not reveal file content
Provider-managed SSE	At rest inside the storage infrastructure only	Managed by the storage provider	Simple to run; single encryption layer provides a smaller security envelope than a layered model

Models summarized for comparison; Everabyte implements the layered model in row three.

Everabyte deliberately implements the layered model. The reasoning is honest and follows from Chapter 1: a server-side model lets Everabyte's services process exactly the operations you authorize - uploads, restoration, version integrity - without pretending the platform never touches your content during those operations, while still making direct access to stored objects produce only ciphertext.

Two things the layered model does not claim: it is not zero-knowledge storage - Everabyte's application services apply the first encryption layer within the platform - and it is not client-side encryption for end users. If your compliance environment requires strictly client-side encryption, a different tool is the right answer; the honest buyer's question is always "what does the platform do with the data it processes?"

What stays yours, in every model

Owner-controlled decryption is the non-negotiable: your decryption key remains under your control - Everabyte services cannot use it or access decrypted stored files without an operation authorized by you. Your recovery passphrase is the backup mechanism for that key, covered in Chapter 3.

Everabyte Encryption Architecture

Three engineering components cooperate across the architecture: the layered encryption context on stored files, the Vault-based management of infrastructure master keys, and the authorization gate in front of every decryption operation.

THE ARCHITECTURE IN FOUR COMPONENTS



Infrastructure master keys — Vault

Everabyte's infrastructure master encryption keys are managed through secure Vault-based key management - the operational standard for rotating, wrapping and separating responsibilities between machine-managed keys and human-accessed secrets. Master keys are distinct from the account-controlled decryption keys and recovery passphrases covered below.

Owner-controlled decryption — the authorization gate

Your decryption key remains under your control: Everabyte services cannot use it or access decrypted stored files without an operation authorized by you. One clarification that reviewers rightly probe: during an authorized upload, Everabyte services do process your content — that is what applying the first encryption layer means — and this engineering reality is what makes post-upload integrity restorable. What Everabyte does not do is access decrypted stored files without an operation you authorize.

Recovery passphrase — the mechanism that keeps you in charge

Your recovery passphrase enables you to restore your encrypted files through Everabyte's secure recovery process. That is its only role, and the way it is treated matters operationally:

- Keep it secure, and never treat it as a user password to be reused anywhere else.
- Never share it with team members or support — Everabyte's secure recovery process is designed to restore access, not to require shared secrets.
- Store it offline, in whatever channel your organization already trusts for key material.

Direct access vs. authorized processing

Direct access to stored objects does not reveal the original file contents - that is what two encryption layers buy. Authorized operations (upload, restore, version verification) are the only path where content processing happens, and they remain yours to initiate, gate and audit.

Performance Considerations

Layering a second encryption step into storage can be cheap in practice: modern CPU cryptographic acceleration makes AES-GCM operations compute-heavy on paper but light at the storage path, which is exactly why layered server-side models are workable at scale. This chapter stays qualitative — we promise no invented throughput numbers.

Operationally, the layered model gives customers three concrete operational advantages:

- No client-side cryptography dependency: end-user devices pay no encryption overhead, which keeps client software simple, stable and identical across platforms.
- Ciphertext at rest from the point of write: application-level encryption happens before files are written, so no post-write window is left in which the first layer is not yet protecting file content.
- Bandwidth scaling options for growing workloads: upload and download behavior is engineered without assuming unmeasurable speed claims - verify any performance-sensitive workload under your own conditions before committing.

How to benchmark us honestly

Run a representative workload for your data sizes with the encryption path active, from your network. Any performance-sensitive projection in a sales conversation should be reproducible with this direct method - and Everabyte's team will help you set the test, because a measurement you run yourself is the only credible one.

Storage Regions and Residency

Encryption decides whether your stored files are readable; your region decides where the storage infrastructure that holds them sits - and region choice is part of the deployment plan for compliance, not a cosmetic configuration flag.

THE REGION MODEL, IN FOUR NUMBERS

12

Storage regions across three continents

1

Storage region assigned per environment

0

Automatic cross-region copies managed by Everabyte

London

Default region for individual accounts

- Each object resides in its assigned storage region — without automatic cross-region replication managed by Everabyte.
- Individual accounts default to London.
- Region choice is available to eligible Company owners with an active EverStream or EverBlaze plan, under the product rules in effect.
- Keep the distinction clear: storage region placement is separate from the location of account data, billing

records and activity logs.

What "residency" answers for the security review

"Where does our data live, and what encryption surrounds it?" Both answers are documented configuration: a chosen region (or the London default for individuals) and the layered encryption of Chapters 1 and 3 applied in that region. Compatible answers are significantly easier to document than a distributed and unmeasured topology.

CHAPTER 06

Compliance Mapping (GDPR)

GDPR Article 32 asks data controllers and processors to account for the technical measures protecting personal data. Everabyte's encryption controls map onto that list in a way that is directly relevant to GDPR review — with the caveat that this document is security-engineering mapping, never legal advice.

Article 32 theme	Everabyte control that answers it	How it holds
"Pseudonymisation and encryption of personal data..."	Two-layer encryption: application-level AES-256-GCM before storage + at-rest encryption inside the storage infrastructure	Stored objects reveal no content on direct access; ciphertext is the default at rest
"...a process for regularly testing, assessing and evaluating..."	Restore-path verification discipline (companion paper, Chapter 5)	Protective state is measured by testing, not assumed by policy
"The ability to ensure ongoing confidentiality..."	Owner-controlled decryption behind an authorization gate	Every decryption operation runs against an action you initiated
Integrity and availability of systems and services	Object versioning, snapshots, retention policies (where enabled)	Accidental or malicious modification leaves prior states recoverable

Reading: the extent of Article 32's "availability and resilience" requirement is a controller-level choice layered on top of the platform's capabilities.

What this mapping does not do

It does not make a legal determination of controller/processor roles, and it does not certify any framework. Share this document with your DPO or counsel; the engineering answers here hold up under Article 32 review, but the responsibility of scoping them stays with you.

Implementation Guide

Adopting layered encryption is mostly an authentication and operations question. Five steps take a new deployment from default to verified:

- Confirm the encryption defaults: application-level encryption on each write path, storage-level at rest, and TLS 1.3 on connections. Each is on by default and should stay on.
- Own the decryption keys consciously: the account-controlled key structure is the first thing your security reviewer wants mapped - its custody matters more than the algorithm string.
- Handle the recovery passphrase correctly: store it offline, restrict knowledge to authorized holders, never treat it as a login credential.
- Test your restore path under your own conditions: a restore exercise in a staging account tells you more than any brochure number.
- Record where your data lives: capture the assigned region (or London default) in your compliance documentation, together with the two-layer encryption description from Chapter 3.

The honest sequencing

Step 4 and step 5 are where most organizations either win or lose their security review. Encryption configuration is inherited automatically from the platform; region documentation and a tested restore path are the only number you get to be measured against.

Free Audit Checklist

Run this checklist against every storage provider you are evaluating - including Everabyte. Tick only where the answer is documented configuration, not a sales claim.

BEFORE-YOU-BUY CHECK

- | | |
|---|---|
| ☐ Encryption schema documented in writing (layers, algorithms, key roles) | ☐ Restore-path test executed in a live or staging account |
| ☐ Tells you plainly whether it is zero-knowledge, client-side or server-side | ☐ Region for each environment documented, single region per environment defaults |
| ☐ Recovery mechanism defined with a real process, not a promise | ☐ Retention, versioning and snapshots available where your data requires them |
| ☐ Decryption custody rules explicit - authorization gate on every decryption run | ☐ Terms, security architecture and privacy pages consistent with the checklist |

The comparison is the point: run the same checklist against your incumbent provider and against Everabyte. Any answer that can only be described as marketing should read as a “no” - and a checklist this simple can be completed in roughly 15 minutes in a vendor's documentation, which is exactly why we

publish it.

WHAT THIS PAPER DOES AND DOES NOT CLAIM

- Does: describe Everabyte's encryption model as it is built today, with no invented throughput or industry statistics; map controls to GDPR Article 32 themes
- Does not: claim zero-knowledge or client-side encryption; claim any certification; give legal advice on controller/processor roles

About this document

This whitepaper is an engineering document: it describes Everabyte's encryption architecture as built and documents the terminology that customer security reviews run on. Companion documents on everabyte.com:

- Security architecture - the full protection model, drawn layer by layer, at everabyte.com/en/security-architecture.
- S3 Security Reality Check 2026 - independent bucket-census research and the 15-minute hardening checklist (public/pdf/Everabyte_S3_Security_Reality_Check_2026.pdf).
- The Immutable Storage Revolution - retention, Object Lock and restore-path verification (public/pdf/Everabyte_Whitepaper_Q1_2026.pdf).
- Privacy policy and terms of service - the contractual scope of everything above, at everabyte.com/en/privacy-policy and everabyte.com/en/terms-and-conditions.

KEEP GOING

Know exactly what your storage does with your files.

» **Security architecture**

<https://everabyte.com/en/security-architecture>

» **Whitepapers and technical reports**

<https://everabyte.com/en/resources/whitepapers>

» **EverStream object storage**

<https://everabyte.com/en/products/everstream>