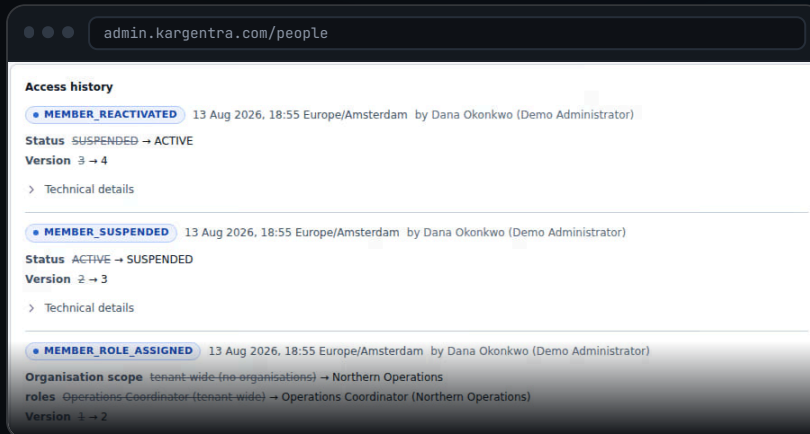


● TRUST & SECURITY

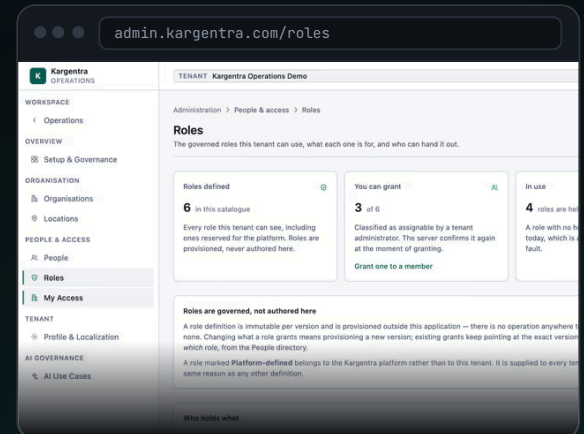
Trust & Security

Logistics software security, isolation and evidence you can verify

Kargentra treats logistics software security as something the system proves: “who could see this, who changed it, and can you prove it?” is answered by the product itself. Tenant data is isolated by the database, access is decided in separate layers, sensitive actions leave hash-verifiable evidence, and Ask Kargentra reads only what the person asking is allowed to see.



ADMINISTRATION · ACCESS CHANGE AUDIT TRAIL WITH BEFORE AND AFTER VALUES



ADMINISTRATION · GOVERNED ROLES AND THE PERMISSIONS EACH GRANTS

WHAT CHANGES FOR THE OPERATION

Isolation enforced by the database

Tenant tables run under forced row-level security beneath application checks, so another tenant's record reads as not found even if a query is wrong.

Access you can explain

Entitlement, permission, record scope, workflow state and screen visibility are checked separately, and every permission traces to a role and an API contract.

Evidence, not assertions

Actions are written to a hash-chained audit record with the actor, their membership and a correlation id, and integrity can be checked in the product.

AI that proposes, people who approve

Ask Kargentra reads permitted context only, cites its sources and proposes. A named role approves, every invocation is recorded and nothing is sent externally on its own.

WHAT TRUST & SECURITY DOES

Tenant isolation with row-level security

Every tenant-owned table runs under forced row-level security for a non-owner runtime role that must carry a tenant context. It is the last line beneath controller authorisation, not the only one.

- Another tenant's record returns a non-disclosing not found
- New tables ship with row-level security forced from the start
- Isolation tests run in every connected release

Layered authorisation

Entitlement is not permission, and permission is not record scope. Each layer is enforced on its own: what the tenant bought, what the role allows, which records the person may act on, whether the transition is lawful, and what the screen shows.

- One availability decision drives the API, preview and navigation
- Calls that bypass the screen are refused by the API
- Unknown facts render inert, never as a working link

Governed roles and My Access

Every API operation names the permission it requires, and only a versioned role can grant it. Membership has a governed lifecycle, and My Access lets each person check what they can do and why.

- Warnings when access is unexplained or survives removing a role
- Platform roles are never tenant memberships
- Platform staff have no access to tenant business data

Hash-verified audit evidence

Business and administrative actions are written to a tenant-scoped audit record where each entry carries the hash of the one before it, with actor, membership and correlation id.

- SHA-256 record hash chained to the previous hash
- Document access records authorisation, never a claimed download
- Configuration changes browsable in Change History

Scoped, revocable shipment sharing

A shipment's owner grants a named colleague a read-only share with a scope, a purpose and an optional end date. The recipient sees only the granted facts, and revocation takes effect on the next request.

- Scopes: shipment status and cargo summary
- Purposes: operational coordination, customer visibility, claims review
- Access can end automatically within 365 days

Sealed, replaceable credentials

Each integration credential is encrypted with its own AES-256-GCM data key, wrapped by a key-encryption key and bound to the tenant and connection, so a copied row cannot be decrypted elsewhere.

- Values are never shown again after storing
- Rotation proven by a keyed fingerprint
- Named refusals when the key is missing or malformed

HOW IT FLOWS

How every request is authorised and recorded

- 01 Identity verified**
The request carries a token for a named person or machine client, verified against a single expected audience.
- 02 Tenant context bound**
The person's chosen workspace binds the database session to one tenant, and row-level security is forced from there.
- 03 Entitlement checked**
The product and module must be in the subscription, switched on and owned by Kargentra in the process route.
- 04 Permission checked**
The person's role must grant the permission the API contract names for this operation.
- 05 Record scope checked**
The person must be able to act on this organisation, site or customer's record, or hold an active share.
- 06 Transition checked**
The change must be lawful from the record's current workflow state, or it is refused with a reason.
- 07 Evidence written**
The action lands in the hash-chained audit record with actor, membership and correlation id.

ASK KARGENTRA IN TRUST & SECURITY

"Who can see shipment SH-61207?"

Shipment SH-61207 has one active share: a warehouse supervisor at the Riverside site holds the Shipment status scope for operational coordination, ending on 31 October. A second share was revoked on 7 October, and a read attempt by that recipient on 8 October was refused and recorded. All 14 evidence records match their recorded hash.

SHIPMENT SH-61207 · SHARING

SHARE GRANT · ACTIVE · ENDS 31 OCT

EVIDENCE · REFUSAL 8 OCT

INTEGRITY CHECK · 14 OF 14 MATCH

PROPOSAL · APPROVED BY THE SHIPMENT OWNER

Revoke the remaining share now, because the coordination purpose ended when the shipment was delivered.

Ask Kargentra reads only what your role can open, cites every fact and proposes. A named person approves; nothing is sent or executed on its own.

EXCEPTIONS AND EDGE CASES HANDLED

- A request for another tenant's record returns not found, without confirming that the record exists.
- A share revoked while the recipient is reading ends on their next request, and the refusal is recorded.
- An expired share is replaced atomically, and a revoke sent twice at once takes effect only once.
- An evidence history that no longer matches its recorded hashes is flagged as a hash mismatch.
- A credential copied to another tenant or connection fails to decrypt, because the ciphertext is bound to both.
- A missing or malformed key-encryption key produces a named refusal rather than a fallback to plain text.

CAPABILITY REFERENCE

Tenant isolation

- Forced row-level security on tenant-owned tables
- Non-owner runtime role requiring tenant context
- Non-disclosing not found across tenants
- Isolation tests in every connected release

Access control and identity

- Entitlement, permission, record scope, workflow state and visibility checked separately
- Permission named on every API operation
- Versioned roles and governed membership
- My Access explainer with warnings
- One OIDC realm across every portal and app
- Machine clients and revocable driver device enrolment

Audit and evidence

- Hash-chained audit records (SHA-256)
- Actor, membership and correlation id on every record
- Document access audit at authorisation
- Short-lived presigned document reads, never stored
- Versioned documents with supersession
- Change History for configuration

Scoped sharing

- Owner-granted, read-only, same-tenant shares
- Two scopes and three purposes
- Optional end date up to 365 days
- Revocation effective on the next request
- AI context opt-in, gated by the workspace switch
- Evidence history with integrity check

Credentials and diagnostics

- AES-256-GCM per-credential data keys
- Key-encryption-key wrapping with re-wrap on rotation
- Ciphertext bound to tenant and connection
- Metadata-only request log from a closed allowlist
- Correlation-id troubleshooting with re-authorisation
- Problem Details errors without internals

AI governance

- Reads only permitted context
- Cites every fact it uses
- Proposals applied only by a named approver
- Every invocation recorded
- Tenant control per capability and autonomy mode
- No automatic external send

STANDARDS AND CONNECTIONS

OIDC SINGLE SIGN-ON

MACHINE-TO-MACHINE CLIENTS

S3-COMPATIBLE OBJECT STORAGE

HMAC-SIGNED WEBHOOKS

PROBLEM DETAILS ERROR CONTRACT

QUESTIONS BUYERS ASK

How is tenant data isolated?

Every tenant-owned table runs under forced row-level security in the database, for a runtime role that must carry a tenant context, beneath the application's own authorisation checks. A request for another tenant's record returns not found without confirming it exists. New tables ship with row-level security forced, and isolation tests run in every connected release.

What is recorded in the audit trail?

Business and administrative actions are recorded with the actor, their membership, the time and a correlation id that links back to the request. Each record carries the hash of the previous one, so an altered or removed entry shows as a mismatch. Access changes keep before and after values, and document access is recorded when it is authorised.



See Trust & Security on your own scenario

Bring a real lane, site or exception. More at www.kargentra.com/trust/

info@kargentra.com

WWW.KARGENTRA.COM · SIGN IN: APP.KARGENTRA.COM