ProjectX

SECURITY WHITEPAPER

What we will not do, and what stops us.

Thirteen properties the platform holds, the compliance frameworks we are working towards and where each actually stands, and seven boundaries on what this technology may be used for. Stated with the mechanism that enforces each one, because a commitment nobody can check is not a commitment.

One decision point for every permission

Every service asks the same policy decision point whether an action is allowed, and every answer carries its reason. There is no second, quieter path to the data.

One question, asked everywhere
Each service authorizes each action through a single decision endpoint, against one catalog of permissions in the form pillar:type:action.

Every no has a reason
A denial returns the reason and the policy that made it, so a refusal can be explained rather than guessed at.

Closed when in doubt
If the decision point cannot be reached, the answer is no.

Ask it questions
Simulate a change before it ships, ask who can act on a resource, or ask what a principal can do.

A tamper-evident audit

Every allow and every deny, from every pillar, lands in one hash-chained record that can be verified, exported and kept under legal hold.

Allows and denies, in one schema
Audit events from every service share one schema and reach the platform chain durably, including the decisions that refused something.

Chained, and checkable
Each event is hash-chained to the one before, with a verification endpoint, and the chain can optionally be anchored externally.

Kept, queried and handed over
Retention respects legal minimums and legal hold; events are queryable by principal, resource, action, time and decision, and export to CSV, Parquet, OCSF and CEF.

Residency before placement

Where something may run is decided before where it will run. Residency rules filter the candidates first, so placement can only ever choose a compliant one.

Filter first, then choose

Residency obligations remove offerings from consideration before the placement resolver picks among the rest. A non-compliant choice is never made and then corrected.

A policy, not a contract clause

Residency is set per tenant and per data classification, and enforced as a substrate allow-list at the policy decision point.

It follows the data's sensitivity

Because a residency policy can be set per classification, the most sensitive data can be held to a narrower set of places than the rest of the tenant.

The invariants

1. One interface for every kind of thing

A table, a camera, a server and a person are the same kind of object to the platform, with one identifier grammar and one authorization call. Nothing gets its own private path.

Held by: One URN grammar, mirrored in TypeScript and Python, with a test that the two agree exactly.

2. One decision point

Exactly one component decides whether you may do something. No service works it out for itself, and there is no superuser flag or wildcard service account — including for us.

Held by: The `projectx/no-local-authz` lint rule refuses anything that looks like a local permission check, everywhere except the policy engine itself.

3. Deny by default, and fail closed

No matching policy means no. If the thing that decides is unreachable, the answer is no — never yes. An outage cannot become an opening.

Held by: Cedar semantics: no match denies, and any forbid beats every permit. Tests assert a 503 rather than a pass when the decision point is down.

4. Everything is audited, including refusals

Every action is written to a tamper-evident log — the ones that succeeded and the ones that were refused. No route opts in, and none can opt out.

Held by: Shared middleware writes the events; the `projectx/no-direct-audit-write` rule stops any pillar writing its own and breaking the chain.

5. One usage envelope for all five pillars

Usage from a server, a bucket and a camera is recorded in the same shape, so one bill reconciles and cost can be attributed without caring which pillar produced it.

Held by: A single metering contract every pillar emits against, with one spelling per meter.

6. If it exists, it is in the catalog

Every resource the platform manages is registered, so "what do we actually have" is a question with an answer. A resource that is not in the catalog is one nobody can find, cost or govern.

Held by: Registration is part of creating a resource, not a later step; a pillar that stops registering is a catalog gap the reindex runbook treats as that pillar's bug.

7. Content stays in the data plane

Video frames, table rows and file contents never enter the control plane. What crosses that line is counts and identifiers — so an audit trail or a support ticket cannot become a copy of your data.

Held by: The control plane has no route that returns content. Runbooks say so explicitly: a troubleshooting ticket is not an exception.

8. Another tenant's data is absent, not forbidden

You cannot see what belongs to somebody else, and you cannot detect that it exists either. There is no default tenant and no cross-tenant read.

Held by: Requests carry a tenant that must match the caller; the `projectx/no-default-tenant` rule refuses any fallback value, so a missing tenant throws rather than guessing.

9. Residency is decided before anything is placed

Where your data may live is settled by policy before a provider is chosen, not checked afterwards. Provisioning outside your region is blocked, not flagged.

Held by: A region restriction evaluated at the decision point, ahead of the driver — so the placement code cannot put work somewhere the decision did not allow.

10. Entitlements are applied when the query is planned

What you may see is built into the query, not filtered out of the results. Two people asking the same question get two different and correct answers, and the shorter one says how much was withheld.

Held by: Row and column filters are injected into the query plan. A lint rule refuses filtering by decision after a fetch, and tests assert the recorded call order.

11. Every provider sits behind an internal interface

No provider's vocabulary reaches the rest of the platform, so moving a workload between clouds — or onto hardware you own — is a substrate change rather than a re-integration.

Held by: The `projectx/no-provider-sdk-above-driver` rule refuses a provider SDK or a named driver above the boundary, and refuses upstream names on customer-facing output.

12. The edge keeps working without us

An on-site node runs for a day with the control plane unreachable and reconciles when the link returns. Recording does not stop because a network did.

Held by: A local buffer sized against the audit spool, and disk pressure that expires the oldest unheld clips first while preserving anything under legal hold.

13. We take no margin on hardware you own

On a substrate you bought, the infrastructure cost and our margin are both zero, and we hold no standing access to it. Break-glass on your hardware needs your own accountable owner to approve it.

Held by: The rating code forces both to zero on a customer-owned substrate even when handed a rate, and the substrate registry carries who owns and who operates each one.

Compliance, and where it stands

Framework	Target	Where it stands
ISO 27001, 27017, 27018	Certification path from year one	Not certified. Controls implemented and evidenced; no audit performed.
SOC 2 Type II	Year one to two	Not certified. No observation period has run.
GDPR	Day one, by design	Article 30 records, DPIA support and deletion certificates are generated from the platform rather than assembled by hand.
NIS2	Design-aligned for critical infrastructure customers	Design-aligned. Alignment is our assessment, not a third party's.
ISO 27701 privacy	Year two	Not started.
PCI DSS	Where financial customers require it	Not in scope today. No cardholder data is processed by the platform.
Sovereign-cloud schemes (C5, SecNumCloud-style)	Per target market	Not started. Air-gapped and customer-owned deployment exist as the technical basis.
FIPS 140-3 validated crypto	Where government customers require it	Not validated. External HSM and KMS binding is in progress.
CIS benchmarks	Cloud compliance scanning	Not yet reported. Benchmark scanning across the fleet is in progress; the results are not yet published per workload.
WCAG 2.2 AA	Every console	Automated checks run over every console and this site on every commit, plus a contrast gate. The screen-reader audit by a person is arranged and has not yet happened.

Responsible use boundaries

1. Face recognition is off by default and separately licensed

Biometric watchlist matching is a separate module, disabled unless three things exist: a contractual addendum, a named accountable owner in your organization, and a recorded lawful basis. All three, or it does not turn on.

Enforced in software

2. Purpose limitation is enforced in software

Searching the archive beyond a configurable window requires a stated purpose, which is logged and reportable. Not a checkbox — the search does not run without it.

Enforced in software

3. Every search and every match is audited

Operator, query, scope and result count, for every search. Your own administrators cannot switch video audit logging off — only shorten retention within legal minimums.

Enforced in software

4. Anonymized analytics mode is available

Counting people without identifying them is the mode most deployments should use, so it has to exist as a mode rather than as advice.

Enforced in software

5. Redaction on export is the default

An export requested with no redaction argument at all comes back redacted. Not an option somebody has to remember under pressure.

Enforced in software

6. We do not sell for mass surveillance

No sale where the stated use is mass surveillance of a general population, suppression of lawful assembly, or targeting of protected groups. A deal-blocker in qualification, not a preference.

Enforced in contract, pinned by test

7. Deletion is real

Retention expiry produces a deletion certificate, and there is no shadow copy kept somewhere else. Deleted means gone, and you get the paperwork.

Enforced in software