THINKING OS™

SEAL Legal Runtime External Proof Packet

Wrong-Authority Filing Risk Before It Leaves Your Firm

Controlled evaluator tests showing what SEAL approved, refused, or routed through authorized supervision before the filing path proceeded.

Public • Redacted • v4.0 • August 2026

Doc ID: TOS-SEAL-EPP-v4.0-7j5k6l

Document: SEAL External Proof Packet
Version: v4.0 (Public • Redacted) – August 2026
Owner: Thinking OS™
Use: May be shared externally. For information only; not legal advice.

How to Use This Packet

Leadership summary — pages 3–15
The claim, what was tested, what it does and does not prove, where SEAL sits, ownership, and legal-judgment boundaries.

Scenario walkthrough — pages 16–27
Showing approval, refusal, missing authority conditions, and authorized supervision.

Technical evidence — pages 28–51
Redacted artifacts, runtime evidence, downstream alignment, scoped-path testing, provenance, and evaluation boundaries.

Table of Contents

The One Claim This Packet Tests 3
Proof in Plain English — Five Things We Tested 5
What This Does / Does Not Prove 8
Why Existing Controls Stop Before the Firm Is Committed 10
Who Owns What 14
What SEAL Does Not Do 15
Scenario 1 — Authorized Filing 16
Scenario 2 — Wrong Authority 18
Scenario 3 — Role Not Permitted 20
Scenario 4 — Required Consent Missing 22
Scenario 5 — Authorized Supervised Path 24
What the Scenarios Show Together 26
Technical Evidence Begins Here 28
- **Approval Evidence** 30
- **Wrong-Authority Refusal Evidence** 34
- **Role-Not-Permitted Refusal Evidence** 36
- **Missing-Consent Refusal Evidence** 38
- **Authorized Supervised Path Evidence** 40
- **Decision and Downstream Result Alignment** 44
- **Could the Demo Workflow Skip SEAL?** 47
- **Proof Provenance / Redactions / Evaluation Boundary** 50

Doc ID: TOS-SEAL-EPP-v4.0-7j5k6l

Document: SEAL External Proof Packet
Version: v4.0 (Public • Redacted) – August 2026
Owner: Thinking OS™
Use: May be shared externally. For information only; not legal advice.

The One Claim This Packet Tests

This packet tests one narrow question:

Can SEAL evaluate one filing action before it leaves the firm, return a governed authority decision, and leave a reviewable record of what it decided?

That is the claim.

Not whether SEAL can practice law.

Not whether the filing is legally correct.

Not whether every workflow in a law firm is governed.

Not whether this packet proves buyer-specific production readiness.

For the evaluator-controlled legal workflow shown in this packet, SEAL is tested at one final-submit boundary before the filing path proceeds.

What the tests show

Under the configured evaluator conditions:

- **Authorized filing conditions produced approval.**
- **Wrong authority produced refusal.**
- **A role that was not permitted for the action produced refusal.**
- **Missing required consent produced refusal.**
- **A refused action could change only through a separately governed supervised path with authority recorded.**
- **For the configured demo path, a direct attempt without the required governed SEAL outcome was rejected.**

Each governed outcome produced reviewable decision evidence.

What this means in plain English

The firm defines the authority rules.

SEAL does not decide legal strategy, legal merits, ethics, or whether a filing is wise.

SEAL answers a narrower question:

Is this actor authorized to take this action, in this matter, under this authority, before the filing leaves the firm?

The packet then shows the evidence behind that answer.

What this packet does not prove

This packet does **not** prove:

- that any specific law firm policy is legally correct;
- that every workflow in a buyer environment is governed;
- buyer-specific production readiness;
- production non-bypassability in a future customer environment;
- that SEAL replaces lawyer judgment, supervision, court rules, IAM, GRC, matter systems, DMS, or filing tools;
- that any buyer has deployed SEAL in production.

The exhibits are **synthetic, evaluator-controlled tests** shown with public redactions.

The proof point is narrow: SEAL can make and record an authority decision before the scoped filing path proceeds, and a qualified evaluator can inspect the resulting evidence.

Proof in Plain English — Five Things We Tested

The easiest way to read this packet is to start with the outcomes.

Thinking OS™ used **synthetic, evaluator-controlled legal filing scenarios** to test whether SEAL would make different authority decisions when the facts changed.

Here are the five core tests.

1. The filing met the required authority conditions

What happened:
The filing request had the required role, authority, consent, matter, and policy conditions.

What SEAL did:
Approved the governed action.

What the test shows:
SEAL does not refuse everything by default. When the configured authority conditions are satisfied, the governed path can proceed.

---

2. The actor had the wrong authority for the filing

What happened:
The actor held a senior legal role, but that role was not authorized for the specific filing action under the configured policy.

What SEAL did:
Refused the action.

What the test shows:
Seniority is not authority.
SEAL evaluates whether the actor is authorized for the specific governed action, not whether the actor has an impressive title.

3. The actor’s role was not permitted to take the action

What happened:
The actor was correctly identified as a paralegal, but that role was not permitted to take the governed filing action.

What SEAL did:
Refused the action.

What the test shows:
Knowing who someone is does not automatically mean they are authorized to take every action available in the workflow.

4. Required consent was missing

What happened:
The actor, filing type, and timing were otherwise acceptable, but a required consent condition was not satisfied.

What SEAL did:
Refused the action.

What the test shows:
A filing can fail the authority check even when the actor and workflow otherwise appear valid.

5. A refusal could change only through an authorized supervised path

What happened:
The original governance decision was refusal. A supervising authority then used the configured supervised path with the required authority evidence.

What SEAL did:
Preserved the original refusal and recorded the authorized supervised outcome.

What the test shows:
Supervision is not a silent bypass. The original “no” remains visible, and the later authorized decision is separately recorded.

What all five tests have in common

In every scenario, the question stayed the same:

Was this actor authorized to take this action, in this matter, under this authority, before the filing path proceeded?

The facts changed.

The governed outcome changed with them.

And each outcome left reviewable decision evidence.

These are controlled evaluator tests, not customer production deployments.

The pages that follow show the underlying artifacts and technical evidence supporting these plain-English summaries.

Disclaimer: *For information only. Not legal advice. This packet is a public evaluation reference for one narrow governed workflow.*

What This Does / Does Not Prove

This packet is intentionally narrow.

It is designed to show what happened when **SEAL Legal Runtime was tested against one controlled legal filing workflow under configured evaluator conditions.**

The exhibits should be read as evidence of that scoped control behavior — **not as proof of every surrounding legal, technical, or production condition.**

What This Packet Does Prove

For the evaluator-controlled workflow shown here, the evidence demonstrates that SEAL can:

- evaluate a filing action before the configured filing path proceeds;
- return different authority outcomes when the underlying facts change;
- approve when the required authority conditions are satisfied;
- refuse when the actor, role, or required authority condition does not satisfy the configured posture;
- preserve an original refusal when a later authorized supervised path changes the final outcome;
- produce reviewable decision evidence for the governed outcome;
- show alignment between the governed decision and downstream handling where that evidence is included;
- reject a direct attempt outside the required governed path in the configured demo workflow.

In plain English:

The tests show that SEAL can make an authority decision before the scoped filing path proceeds — and leave evidence of what it decided.

What This Packet Does Not Prove

This packet does **not** prove:

- that any law firm's policy or authority model is legally correct;
- that a filing is legally sufficient, strategically wise, ethical, or appropriate in a particular jurisdiction;
- that upstream identity, matter, consent, or other source data is accurate;
- that every workflow or execution path in a buyer environment is governed;
- production non-bypassability inside a future customer environment;
- buyer-specific security, integration, availability, or production readiness;
- that a refusal prevented malpractice, financial loss, legal harm, or another downstream consequence;
- that SEAL replaces lawyer judgment, professional supervision, court rules, IAM, GRC, matter systems, DMS, filing tools, or other firm controls;
- that any buyer has deployed SEAL in production.

Why That Boundary Matters

A serious proof packet should make clear **where the evidence stops**.

These exhibits are not a claim that SEAL governs an entire firm. They **are not** a certification of legal correctness. They **are not** a substitute for customer-specific security, workflow, authority, or production diligence.

They demonstrate a narrower thing:

For the scoped evaluator workflow, the authority control behaved as shown, and the resulting decision evidence can be reviewed.

The tests use **synthetic, evaluator-controlled scenarios** and public redactions. Customer production deployment requires separate scoping, authority mapping, integration review, security review, continuity review, and written agreement.

The packet proves the control behavior shown here. It does not prove the world around the control.

Why Existing Controls Stop Before the Firm Is Committed

Most firms already have important controls around legal work: identity systems, GRC, matter systems, document systems, filing tools, workflow tools, AI gateways, and audit logs.

Those controls matter. But most govern **around** the action: who can log in, what policy says, what document exists, what workflow step occurred, what the AI tool produced, or what happened afterward.

SEAL governs a narrower point:

Before the filing leaves the firm, is this actor authorized to take this action, in this matter, under this authority, with the required evidence?

That is the Commit Layer: the pre-execution authority gate before the institution is committed.

Existing control	What it controls well	Where it stops	Why the Commit Layer is still needed
IAM / SSO	Who can access systems	Does not prove authority for a specific filing or action	Valid login is not valid authority to bind
GRC / policy	Written rules, control ownership, audit posture	Usually not inline at final-submit	Policy must become a runtime verdict
Matter system	Matter, client, workflow context	Does not usually approve or refuse execution	Context must connect to authority before action
DMS	Document access, versioning, storage	Controls the file, not the act of filing or sending	The risk is often the commitment, not the document
Filing / workflow tools	Local process steps and submission flow	Platform-specific, inconsistent across systems	The firm needs one authority surface across workflows
AI gateway	AI interaction, prompt/output/model-call governance	Controls AI interaction, not institutional commitment	Safe output can still drive unauthorized action
Audit / logging	What happened after the fact	Does not by itself own the action-authority decision	A record after the fact is not a pre-commit authority decision.
Commit Layer / SEAL	Runtime approve / refuse / supervised override	Only governs workflows wired through it	Provides an authority decision before the firm is committed

Existing systems govern around the action. The Commit Layer governs the action boundary itself.

What SEAL Governs

SEAL governs one narrow question before a filing leaves the firm:

Is this actor authorized to take this action, in this matter, under this authority?

The firm's existing systems remain the sources of truth for identity, role, matter context, authority, consent, and supervision.

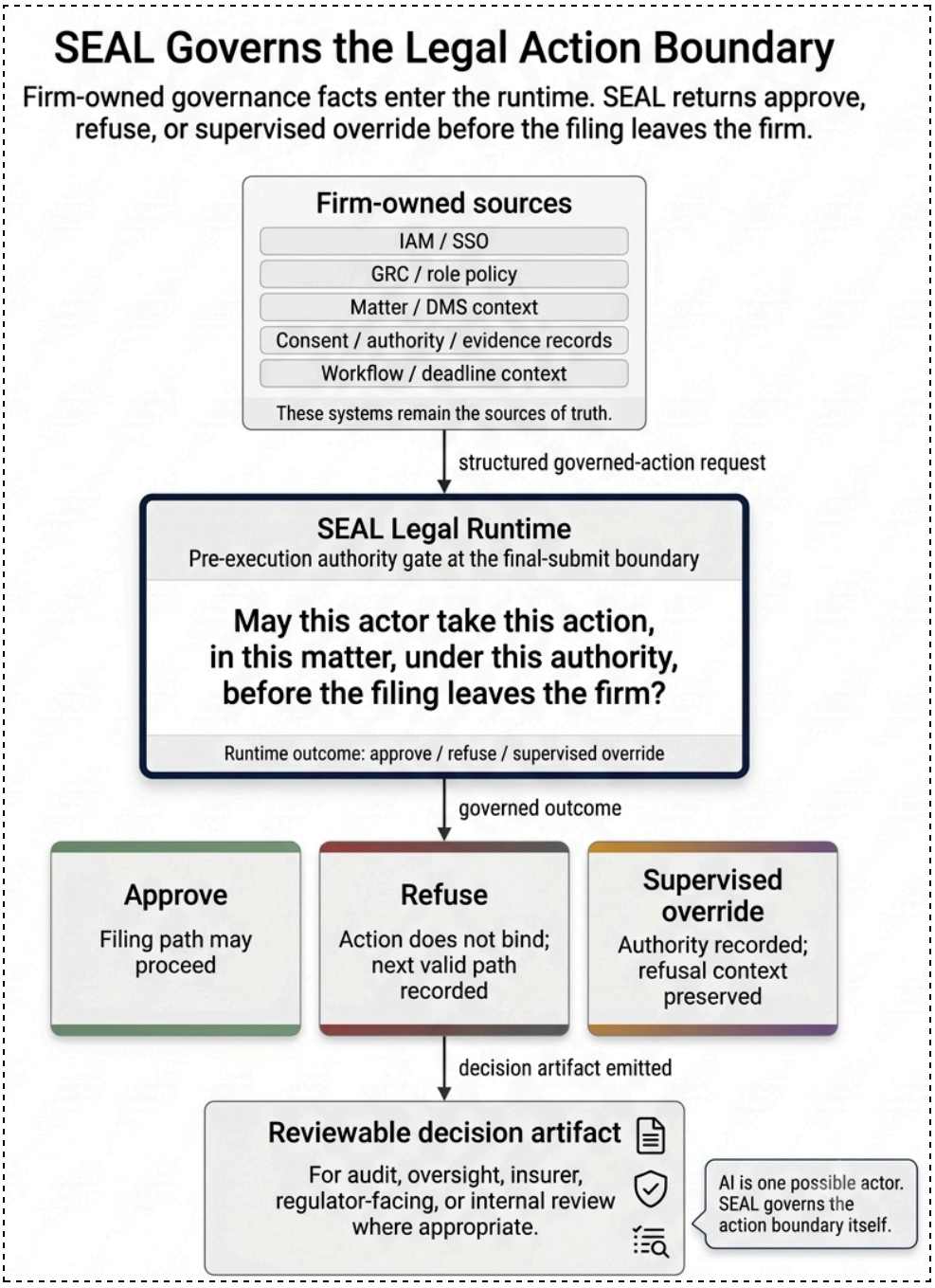
SEAL does not decide whether the filing is legally correct, strategically wise, or appropriate for the matter.

For the scoped workflow, SEAL evaluates the authority conditions and returns a governed outcome:

Approve. Refuse. Or require an authorized supervised path.

Each governed outcome produces a reviewable decision artifact showing what the control decided at the moment of action.

In plain English: SEAL governs whether the action may proceed under the firm's configured authority conditions — not the substance of the legal work.



Your systems provide the governance facts. SEAL evaluates the authority question before the filing leaves the firm. The firm remains responsible for the filing, legal judgment, supervision, and what happens next.

Who Owns What

The Firm Owns

- legal judgment and professional supervision
- policies and authority rules
- identity and role sources
- matter and workflow context
- who may supervise or approve exceptions
- workflow routing and escalation
- retention, use, and disclosure of its decision records
- whether a workflow remains observe-only or later moves into controlled enforcement

Thinking OS™ Is Responsible For

- operating SEAL Legal Runtime within the agreed scope
- returning governed outcomes for the scoped workflow
- producing reviewable decision artifacts
- maintaining the runtime's security, integrity, and availability posture within agreed scope
- supporting the agreed evaluation or integration path

The firm defines the authority model.

SEAL evaluates the governed action against that configured authority posture.

> **The firm owns the rules and legal judgment. Thinking OS™ operates the runtime.**

What SEAL Does Not Do

SEAL Legal Runtime is a **pre-execution authority gate for governed legal actions.**

It evaluates whether the configured authority conditions for the scoped action are satisfied and produces reviewable decision evidence.

It does **not**:

- draft, edit, sign, send, or file legal documents
- provide legal advice
- choose litigation strategy
- decide whether a filing is legally correct or strategically wise
- replace lawyers, supervisors, or professional judgment
- replace GRC, IAM, matter systems, DMS, filing tools, or workflow systems
- become the system of record for the matter or filing
- invent the firm's policies, roles, authority, or supervision model
- govern workflows that are not in the agreed wired scope

SEAL can refuse. It cannot file.

An approval means the configured governance conditions for the scoped action were satisfied.

It does **not** certify legal merits, strategy, ethics, professional judgment, or jurisdictional correctness.

The firm remains responsible for the legal action and what happens next.

Scenario 1 — Authorized Filing

The required authority conditions were satisfied

This test asks the simplest question:

> **What happens when the actor is authorized and the required conditions for the filing are present?**

What happened?

A synthetic, evaluator-controlled filing request was submitted for an **Administrative Law / Motion to Extend Time** workflow.

For this test:

- the actor was recognized as an authorized attorney;
- the required authority conditions were present;
- consent was present;
- the filing request matched the configured workflow and matter context.

What did SEAL decide?

Approve.

Why?

The governed request satisfied the configured authority conditions for that filing action.

SEAL did not approve the legal merits of the filing.

It approved the **authority to take the governed action under the configured conditions.**

Did the action bind?

Yes — in this controlled evaluator test.

After SEAL returned approval, the configured downstream action completed successfully.

The decision evidence and downstream result tell the same story:

approved → allowed to proceed → completed

What happens next?

The firm still owns the legal action.

SEAL does not decide whether the filing is legally correct, strategically wise, ethically appropriate, or ready for submission.

Lawyers and supervisors remain responsible for legal judgment, professional supervision, and the filing itself.

Where is the evidence?

The technical exhibits later in this packet show:

- the SEAL-generated approval artifact;
- the governed approval outcome;
- the configured binding posture;
- the downstream execution result;
- alignment between the governed decision and downstream handling;
- redacted evidence references supporting later review.

Bottom line: *when the required authority conditions were satisfied, SEAL approved the action before the filing path proceeded and left reviewable evidence of that decision.*

Test posture: Synthetic, evaluator-controlled. Public redactions applied. This is not a customer production deployment.

Scenario 2 — Wrong Authority

A senior legal title did not equal authority to file

This test asks:

> **What happens when the actor is known and senior, but is not authorized for this specific filing action?**

What happened?

A synthetic, evaluator-controlled filing request was submitted for an **Administrative Law / Motion to Extend Time** workflow.

The acting role resolved to **General Counsel**.

But under the configured authority rules for this filing path, that role was **not authorized to take the governed filing action**.

What did SEAL decide?

Refuse.

Why?

The actor's title did not match the authority required for that specific filing action.

The important distinction is simple:

> **Seniority is not authority.**

A person can hold a senior legal role and still lack authority for a particular action, matter, or workflow.

Did the action bind?

No — in this controlled evaluator test.

SEAL returned refusal before the configured filing path proceeded.

What happens next?

The firm owns the next valid path.

That may mean correcting the authority context, routing the matter for supervision, or taking another firm-defined action.

SEAL does not choose the supervisor, decide the legal strategy, or determine whether the filing should ultimately be made.

It records the governed authority decision.

Where is the evidence?

The technical exhibit later in this packet shows:

- the SEAL refusal outcome;
- the refusal reason category;
- the configured authority context;
- the decision and policy references;
- redacted evidence and integrity references supporting later review.

Bottom line: *a senior title did not override the firm's configured authority rules. SEAL refused the filing action before the scoped path proceeded*.

Test posture: Synthetic, evaluator-controlled. Public redactions applied. This is not a customer production deployment.

Scenario 3 — Role Not Permitted

The actor was known, but the role was not allowed to take this filing action

This test asks:

> **What happens when SEAL knows who the actor is, but that actor's role is not permitted to take the governed action?**

What happened?

A synthetic, evaluator-controlled legal action was submitted for an **Administrative Law / Motion to Compel Production** workflow.

The actor's identity and group context were recognized.

The acting role resolved to **Paralegal**.

Under the configured authority rules for this workflow, that role was **not permitted to take the governed filing action**.

What did SEAL decide?

Refuse.

Why?

The problem was not missing identity.

SEAL knew who the actor was and which role applied.

The refusal occurred because the **recognized role itself was not authorized for that specific action**.

The distinction is important:

> **Knowing who someone is does not mean they are authorized to do everything the system can reach.**

Did the action bind?

No — in this controlled evaluator test.

SEAL returned refusal before the configured filing path proceeded.

What happens next?

The firm owns the next valid path.

That may mean routing the matter to an authorized lawyer, correcting the workflow, or following another firm-defined supervision or remediation process.

SEAL does not choose the lawyer, decide whether the filing should be made, or replace professional supervision.

It records the authority decision.

Where is the evidence?

The technical exhibit later in this packet shows:

- the SEAL refusal outcome;
- the role-based refusal reason;
- the trusted actor and role context;
- the relevant policy reference;
- redacted decision, evidence, and integrity references supporting later review.

Bottom line: *the actor was known, but the role was not permitted to take this filing action. SEAL refused the action before the scoped path proceeded.*

Test posture: Synthetic, evaluator-controlled. Public redactions applied. This is not a customer production deployment.

Scenario 4 — Required Consent Missing

The actor and filing were otherwise valid, but a required authority condition was missing

This test asks:

> **What happens when the actor and filing context are acceptable, but a required consent or authority condition has not been satisfied?**

What happened?

A synthetic, evaluator-controlled filing request was submitted for an **Administrative Law / Motion to Extend Time** workflow.

For this test:

- the actor was recognized;
- the filing type matched the configured workflow;
- the timing posture was otherwise acceptable;
- but a required **consent condition** was not satisfied at the time of action.

What did SEAL decide?

Refuse.

Why?

The governed filing required consent as part of the configured authority posture.

That condition was missing.

The distinction is important:

A filing can look otherwise valid and still lack the authority required to proceed.

SEAL did not treat the missing condition as a warning or something to reconstruct later.

It returned a governed refusal for the scoped action.

Did the action bind?

No — in this controlled evaluator test.

SEAL returned refusal before the configured filing path proceeded.

What happens next?

The firm owns the next valid path.

That may mean obtaining the required consent, correcting the authority context, routing the matter for supervision, or following another firm-defined remediation process.

SEAL does not decide whether consent should be given or whether the filing should ultimately proceed.

It records whether the configured authority condition was satisfied at the moment of action.

Where is the evidence?

The technical exhibit later in this packet shows:

- the SEAL refusal outcome;
- the missing-consent reason category;
- the relevant policy and authority context;
- the decision reference;
- redacted evidence and integrity references supporting later review.

Bottom line: *the actor and filing were otherwise acceptable, but the required consent condition was missing. SEAL refused the action before the scoped filing path proceeded.*

Test posture: Synthetic, evaluator-controlled. Public redactions applied. This is not a customer production deployment.

Scenario 5 — Authorized Supervised Path

The original refusal stayed on the record, and the action changed only through authorized supervision

This test asks:

What happens when SEAL refuses an action, but the firm has a valid supervised path for an authorized person to review and approve the exception?

What happened?

A synthetic, evaluator-controlled filing request for an **Administrative Law / Motion for Summary Judgment** did not clear the ordinary authority path.

SEAL's original governance decision was:

Refuse.

A supervising partner then used the configured supervised path with documented supervisory authority and the required override evidence.

What did SEAL decide?

The **original refusal remained recorded**.

The final outcome changed only after the authorized supervised path was satisfied.

In technical terms, the test produced a **supervised override approval** linked back to the original refusal.

Why?

The point of supervision is not to erase the original "no."

It is to provide a separately governed path for an authorized person to take responsibility for the exception.

Supervision is not a silent bypass. The refusal remains visible, and the later authorized decision is recorded separately.

Did the action bind?

Yes — in this controlled evaluator test, but only after the authorized supervised path was satisfied.

The action was not treated as an ordinary approval.

The original refusal stayed on the record, supervisory authority was required, and the final governed outcome was allowed only through the configured exception path.

What happens next?

The firm remains responsible for the supervisory decision and the legal action.

SEAL does not decide whether an exception should be granted as a matter of legal strategy or professional judgment.

The firm decides who may supervise, what authority is required, and whether the exception should be used.

SEAL records the governed path and its outcome.

Where is the evidence?

The technical exhibits later in this packet show:

- the original refusal;
- the linked supervised-override outcome;
- the supervising authority record;
- the connection between the override and the original refusal;
- the final binding outcome;
- redacted decision and evidence references supporting later review.

Bottom line: *SEAL did not erase the original refusal. The action changed only through a separately governed supervisory decision with authority recorded.*

What the Scenarios Show Together

The five scenarios test the same narrow authority question under different facts:

> **Is this actor authorized to take this action, in this matter, under this authority, before the filing path proceeds?**

The answer changed when the facts changed.

When the required authority conditions were satisfied

SEAL approved the action.

The configured filing path was allowed to proceed, and reviewable decision evidence was produced.

When the actor had the wrong authority

SEAL refused the action.

A senior title did not override the authority required for the specific filing.

When the actor was known but the role was not permitted

SEAL refused the action.

Identity was not the problem. The recognized role itself was not authorized for that governed action.

When required consent was missing

SEAL refused the action.

The filing context could otherwise be acceptable while still lacking a required authority condition.

When an authorized supervisory path was used

The original refusal remained on the record.

The final outcome changed only through a separately governed supervisory decision with authority recorded.

The Common Pattern

Across all five tests:

- **the firm-defined authority conditions controlled the decision**
- **SEAL returned an explicit governed outcome**
- **the decision occurred before the scoped filing path proceeded**
- **the firm retained legal judgment and supervision**
- **each governed outcome left reviewable decision evidence**

The tests do not show that SEAL decides whether legal work is correct.

They show something narrower:

> **SEAL can distinguish between authorized, unauthorized, and supervision-required filing actions before the scoped path proceeds — and preserve evidence of the decision.**

These are **synthetic, evaluator-controlled tests**, not customer production deployments.

Technical Evidence Begins Here

For Technical Evaluators, Risk, Procurement, Insurers, and Other Qualified Reviewers

The preceding pages explain the five evaluator scenarios in plain English.

The pages that follow contain the **underlying technical evidence supporting those summaries**.

This section is intentionally more detailed.

It includes the actual redacted decision artifacts, evaluator views, runtime outcomes, execution records, evidence references, and configured-path tests used to examine the scoped control behavior.

How to Read This Section

For each exhibit, the evaluator should be able to connect five things:

1. What action was attempted?
The governed legal action and relevant context.

2. What did SEAL decide?
Approve, Refuse, or an authorized supervised outcome.

3. Did the governed outcome bind?
The runtime or enforcement posture applicable to that test.

4. What happened downstream?
Where shown, whether the configured downstream handling matched the governed outcome.

5. What evidence remains?
The decision artifact, trace or decision references, authority and policy context, and supported integrity evidence available for review.

The technical fields exist to support those questions.

They are not separate claims from the plain-English scenarios.

What This Evidence Is Intended to Demonstrate

For the scoped evaluator workflow, the exhibits are designed to let a qualified reviewer inspect:

- governed approval evidence;
- governed refusal evidence across different refusal conditions;
- an authorized supervised path that preserves the original refusal;
- the relationship between governance decisions and downstream handling where demonstrated;
- reviewable decision artifacts and supporting references;
- configured demo-path behavior where a valid governed outcome is required.

Evaluation Boundary

These exhibits come from **synthetic, evaluator-controlled tests** and are shown with public redactions.

They demonstrate the scoped behavior shown in this packet.

They do **not** establish:

- buyer-specific production readiness;
- production non-bypassability in a future customer environment;
- correctness of a law firm's legal or policy judgment;
- universal coverage across unwired workflows;
- replacement of security, integration, operational, or legal diligence.

Public redactions remove sensitive implementation, routing, identity, infrastructure, and other non-public details while preserving the evidence needed to evaluate the stated control behavior.

> **The executive section tells you what happened. This section shows the evidence behind it.**

Approval Evidence

Artifact Format	v1.2
Runtime Decision Contract	v1
Generated By	SEAL Runtime
Runtime Mode	Enforce

Executive Summary — Evaluator View

Display-only hierarchy of upstream governance/runtime metadata. Detailed evidence records follow below.

PLAIN-ENGLISH CONTROL MEANING

Governance approved and required downstream execution completed successfully.

1. Outcome

DECISION
Approved

DID THE ACTION BIND?
Yes

WHY
Governance approved and required downstream execution completed successfully.

2. Governance vs Runtime

Layer	Result
Governance Decision	Approval
Runtime Mode	Enforce
Execution	Attempted / succeeded
Binding Effect	Allowed

3. Authority / Reason

AUTHORITY BASIS
Policy: SEAL-APP-ALLOWED-001 • Policy set: [redacted] • Role: Attorney • Consent: issuer=[redacted], subject=user-123 • Deadline: [redacted], approx. 48.00 hours remaining

CONTROL MEANING
Governance approved and required downstream execution completed successfully.

4. Next Valid Path — Tenant-Owned Handoff

TENANT-OWNED NEXT STEP
Approval and execution are recorded in this artifact. The firm owns post-approval recordkeeping, review, retention, and downstream workflow.

SEAL records the governed approval and execution receipt; the firm owns matter recordkeeping, downstream workflow, review, and retention.

5. Evidence Anchors

ARTIFACT ID
[redacted]

DECISION ID
[redacted]

GOVERNANCE APPROVAL HASH
[redacted]

TIMESTAMP
[redacted]

DECLARED / NOT DECLARED

"Not Declared" means the value was not provided to SEAL at runtime. SEAL did not infer or backfill it.

Evaluator View — Approval Executive Summary

SEAL-generated runtime artifact from a synthetic, evaluator-controlled request, shown with public redactions only. This view below shows the approval outcome, enforcement posture, execution result, binding effect, firm-owned handoff, and redacted evidence anchors. The narrow proof point: when configured authority conditions are satisfied, approval is recorded before the filing path proceeds.

Execution Record — Approved, Enforced, Succeeded, Aligned

Runtime Outcome & Execution Record

Runtime and execution proof record. Values are rendered from upstream runtime/executor metadata; this notifier does not infer approval, execution success, gate status, binding effect, or decision alignment.

Runtime decision path

Governance, pre-execution runtime, final runtime, and alignment facts as supplied by upstream governance/runtime layers.

Governance Decision	approval
Pre-Execution Runtime Decision	approval
Final Runtime Decision	approval
Decision Alignment	aligned

Execution receipt

Receipt-only execution metadata. This notifier renders the supplied execution capsule and does not call or operate downstream systems.

Status	ok
Attempted	True
OK	True
Required	True
Effective Enforcement	True
Execution Outcome	succeeded
Execution Receipt Alignment	aligned
Blocks Effective Completion	False
Control Meaning	Governance approved and required downstream execution completed successfully.

Control meaning

Plain-language runtime/control meaning supplied by upstream runtime or governance metadata when present.

Plain-Language Control Meaning	Governance approved and required downstream execution completed successfully.

This execution view confirms the approval did not remain a policy note or dashboard event. Governance decision, final runtime decision, execution outcome, and execution alignment all point to the same result: approved, enforced, succeeded, aligned. Public redactions remove endpoint, routing, and infrastructure details while preserving evaluator-visible proof of controlled execution.

How to Read This Approval Artifact

This artifact should be read as evidence of runtime authority control, not legal judgment. SEAL did not decide whether the filing was strategically wise, legally sufficient, or jurisdictionally correct. It evaluated whether configured authority conditions were satisfied before the filing path proceeded.

Approval is not the absence of governance. It is a governed commit before the firm is bound.

What to verify

1. **Governed request —** the filing-path action was presented to SEAL before execution.
2. **Approval outcome** — SEAL returned approval under the configured authority posture.
3. **Runtime alignment** — the governance decision and runtime decision matched.
4. **Execution result** — the downstream action completed after the governed approval.
5. **Evidence surface** — reviewable evidence references exist, with sensitive values redacted.

Wrong-Authority Refusal Evidence

Artifact Format	v1.2
Runtime Decision Contract	v1
Generated By	SEAL Runtime
Runtime Mode	Enforce

Executive Summary — Evaluator View

Display-only hierarchy of upstream governance/runtime metadata. Detailed evidence records follow below.

Plain-English Control Meaning

Governance refused this action because Role 'General Counsel' is not authorized for the Administrative Law vertical under the configured role policy.

1. Outcome

DECISION
Refused

DID THE ACTION BIND?
No

WHY
Governance refused this action because Role 'General Counsel' is not authorized for the Administrative Laws vertical under the configured role policy.

2. Governance vs Runtime

Layer	Result
Governance Decision	Refusal
Runtime Mode	Enforce
Final Runtime Outcome	Refusal / did not bind
Decision Alignment	Aligned
Binding Effect	No

Refusal-first outcome: this table separates governance decision, runtime posture, final runtime outcome, alignment, and binding effect.

3. Authority / Refusal Reason

AUTHORITY BASIS
Refusal code: SEAL-ROLE-NOT-AUTHORIZED • Policy set: [redacted] • Role: General Counsel • Deadline: [redacted] approx. 48.00 hours remaining

CONTROL MEANING
Governance refused this action because Role 'General Counsel' is not authorized for the Administrative Laws vertical under the configured role policy.

4. Next Valid Path — Tenant-Owned Handoff

TENANT-OWNED NEXT STEP
Provide authorized role or firm owned review metadata for governed resubmission — Owner: Tenant identity role mapping or supervisory review owner — Required: trusted reviewer identity or trusted claims, legal role authorized for this vertical/action under the configured policy, Firm owned supervisory review metadata when firm policy permits review of this refusal category, resubmission through the governed path only after role authorization, role mapping, or Firm owned supervisory review metadata is accepted by the governed path

SEAL records the governed next path; the firm owns remediation, review, escalation, record-keeping, and workflow.

5. Evidence Anchors

ARTIFACT ID
[redacted]

DECISION ID / CAPSULE ANCHOR
[redacted]

AUDIT TRACE ID
[redacted]

GOVERNANCE REFUSAL HASH
[redacted]

TIMESTAMP
[redacted]

Declared / Not Declared

"Not Declared" means the value was not provided to SEAL at runtime. SEAL did not infer or backfill it.

Scenario

A governed filing request was submitted under the strict runtime path for **Administrative Law / Motion To Extend Time**. The acting role resolved to **General Counsel**, but that role was not configured as an authorized filing role for this governed motion under the tenant's policy.

What SEAL returned

SEAL returned a sealed refusal with code **SEAL-ROLE-NOT-AUTHORIZED**, plus an audit trace reference, policy reference, rule-basis evidence, redacted evidence and integrity references

Why it matters

This shows that SEAL enforces workflow-specific authority, not job-title prestige. Even a senior legal role is blocked when it is outside the firm's configured authorization scope for the action at issue.

Bottom line

Seniority is not authority. SEAL checks whether the actor is authorized for this governed action under the configured rules.

Generated from a synthetic, evaluator-controlled request and redacted for public review. Control outcome and evidence-reference categories are preserved; sensitive values, routing details, operational contacts, and live destinations are redacted.

Role-Not-Permitted Refusal Evidence

Artifact Format	v1.2
Runtime Decision Contract	v1
Generated By	SEAL Runtime
Runtime Mode	Enforce

Executive Summary — Evaluator View

Display-only hierarchy of upstream governance/runtime metadata. Detailed evidence records follow below.

Plain-English Control Meaning

Governance refused this action because Role 'Paralegal' is explicitly disallowed under the Administrative Law vertical policy.

1. Outcome

DECISION
Refused

DID THE ACTION BIND?
No

WHY
Governance refused this action because Role 'Paralegal' is explicitly disallowed under the Administrative Laws vertical policy.

2. Governance vs Runtime

Layer	Result
Governance Decision	Refusal
Runtime Mode	Enforce
Final Runtime Outcome	Refusal / did not bind
Decision Alignment	Aligned
Binding Effect	No

Refusal-first outcome: this table separates governance decision, runtime posture, final runtime outcome, alignment, and binding effect.

3. Authority / Refusal Reason

AUTHORITY BASIS
Refusal code: SEAL-ROLE-DISALLOWED • Policy set: [redacted] • Role: Paralegal • Deadline: [redacted] approx. 48.00 hours remaining

CONTROL MEANING
Governance refused this action because Role 'Paralegal' is explicitly disallowed under the Administrative Laws vertical policy.

4. Next Valid Path — Tenant-Owned Handoff

TENANT-OWNED NEXT STEP
Rerun with authorized role or firm owned review metadata — Owner: Tenant identity role mapping or supervisory review owner — Required: trusted reviewer identity or trusted claims, legal role authorized for this action under the configured vertical policy, Firm owned supervisory review metadata when firm policy permits review of this refusal category, resubmission through the governed path only after role authorization, role mapping, or Firm owned supervisory review metadata is accepted by the governed path

SEAL records the governed next path; the firm owns remediation, review, escalation, record-keeping, and workflow.

5. Evidence Anchors

ARTIFACT ID
[redacted]

DECISION ID / CAPSULE ANCHOR
[redacted]

AUDIT TRACE ID
[redacted]

GOVERNANCE REFUSAL HASH
[redacted]

TIMESTAMP
[redacted]

Declared / Not Declared

"Not Declared" means the value was not provided to SEAL at runtime. SEAL did not infer or backfill it.

Scenario

A governed legal action was submitted under the strict runtime path for **Administrative Law / Motion To Compel Production**. The acting role resolved to **paralegal** from trusted identity and group context.

What SEAL returned

SEAL returned a sealed refusal with code **SEAL-ROLE-DISALLOWED**, plus an audit trace reference, client policy reference, rule-basis evidence, redacted evidence and integrity references

Why it matters

This proves refusal is not limited to unknown identity or missing mapping. Even when identity is known and mapped, SEAL still refuses when the mapped role is structurally barred under the configured policy.

Caption

This exhibit shows a governed refusal produced before the action left the firm. The artifact records who attempted the action, what was attempted, which policy context applied, and why the action was blocked.

Generated from a synthetic, evaluator-controlled request and redacted for public review. Control outcome and evidence-reference categories are preserved; sensitive values, routing details, operational contacts, and live destinations are redacted.

Missing-Consent Refusal Evidence

Artifact Format	v1.2
Runtime Decision Contract	v1
Generated By	SEAL Runtime
Runtime Mode	Enforce

Executive Summary — Evaluator View

Display-only hierarchy of upstream governance/runtime metadata. Detailed evidence records follow below.

Plain-English Control Meaning

Governance refused this action because No valid client consent record or consent token-of-record was declared for this matter and motion.

1. Outcome

DECISION
Refused

DID THE ACTION BIND?
No

WHY
Governance refused this action because No valid client consent record or consent token-of-record was declared for this matter and motion.

2. Governance vs Runtime

Layer	Result
Governance Decision	Refusal
Runtime Mode	Enforce
Final Runtime Outcome	Refusal / did not bind
Decision Alignment	Aligned
Binding Effect	No

Refusal-first outcome: this table separates governance decision, runtime posture, final runtime outcome, alignment, and binding effect.

3. Authority / Refusal Reason

AUTHORITY BASIS
Refusal code: SEAL-CONSENT-001 • Policy set: [redacted] • Role: Attorney • Consent status: missing_or_unknown • Deadline: [redacted] approx. 48.00 hours remaining

CONTROL MEANING
Governance refused this action because No valid client consent record or consent token-of-record was declared for this matter and motion.

4. Next Valid Path — Tenant-Owned Handoff

TENANT-OWNED NEXT STEP
Attach consent record or token for governed resubmission — Owner: Tenant owned matter team or consent system — Required: client consent record or consent token, matching matter / motion scope, resubmission through the governed path with updated consent evidence

SEAL records the governed next path; the firm owns remediation, review, escalation, record-keeping, and workflow.

5. Evidence Anchors

ARTIFACT ID
[redacted]

DECISION ID / CAPSULE ANCHOR
[redacted]

AUDIT TRACE ID
[redacted]

GOVERNANCE REFUSAL HASH
[redacted]

TIMESTAMP
[redacted]

Declared / Not Declared

"Not Declared" means the value was not provided to SEAL at runtime. SEAL did not infer or backfill it.

Scenario

A governed filing request was submitted under the strict runtime path for **Administrative Law / Motion To Extend Time**. The actor, motion, and timing posture were otherwise valid, but the consent condition required for that governed filing was not satisfied at the time of action.

What SEAL returned

SEAL returned a sealed refusal with code **SEAL-CONSENT-001**, plus an audit trace reference, policy reference, redacted evidence and integrity references

Why it matters

This proves refusal is not limited to role or authority mapping. SEAL also refuses when a required consent or authority condition is missing before the filing path proceeds.

Caption

This exhibit shows a governed refusal produced before the filing left the firm. The artifact records who attempted the action, what was attempted, which policy context applied, and why the request was blocked. Refusals are governed runtime outcomes tied to the firm's configured rules and inputs, not post-hoc opinions.

Generated from a synthetic, evaluator-controlled request and redacted for public review. Control outcome and evidence-reference categories are preserved; sensitive values, routing details, operational contacts, and live destinations are redacted.

Authorized Supervised Path Evidence

Artifact Format	v1.2
Runtime Decision Contract	v1
Generated By	SEAL Runtime
Runtime Mode	Enforce

Executive Summary — Evaluator View

Display-only hierarchy of upstream governance/runtime metadata. Detailed evidence records follow below.

PLAIN-ENGLISH CONTROL MEANING

Baseline governance refused the action, but runtime approved it under supervisory override, and required downstream execution completed successfully.

1. Outcome

DECISION
Supervised override — approved
DID THE ACTION BIND?
Yes
WHY
Baseline governance refused the action, but runtime approved it under supervisory override, and required downstream execution completed successfully.

1A. Supervised Override Evidence

PARENT REFUSAL
SEAL-ROLE-NOT-AUTHORIZED • Parent decision: [redacted]
OVERRIDE ACTOR
[redacted]
AUTHORITY RECORDED
Yes
SIGNED AUTHORITY REQUIRED
Yes
SIGNED AUTHORITY RECORDED
Yes
BOUND TO PARENT DECISION
Yes (upstream parent reference supplied)
OUTCOME
Approved through supervised override

Display-only summary of the upstream override_contract. Detailed Formal Override Path record appears below.

2. Governance vs Runtime

Layer	Result
Baseline Governance Decision	Refusal
Runtime Mode	Enforce
Final Runtime Outcome	Approval
Decision Alignment	Diverged
Binding Effect	Allowed

3. Authority / Reason

AUTHORITY BASIS
Policy: SEAL-OVERRIDE-APPROVED • Policy set: [redacted] Role: Partner • Consent: issuer=[redacted] subject=[redacted] • Deadline: [redacted]; approx. 48.00 hours remaining • Override: approved
CONTROL MEANING
Baseline governance refused the action, but runtime approved it under supervisory override, and required downstream execution completed successfully.

4. Next Valid Path — Tenant-Owned Handoff

TENANT-OWNED NEXT STEP
Approval and execution are recorded in this artifact. The firm owns post-approval recordkeeping, review, retention, and downstream workflow.

SEAL records the governed approval and execution receipt; the firm owns matter recordkeeping, downstream workflow, review, and retention.

5. Evidence Anchors

ARTIFACT ID
[redacted]
DECISION ID
[redacted]
GOVERNANCE APPROVAL HASH
[redacted]
TIMESTAMP
[redacted]

DECLARED / NOT DECLARED

"Not Declared" means the value was not provided to SEAL at runtime. SEAL did not infer or backfill it.

Evaluator View — Supervised Override Executive Summary

SEAL-generated runtime artifact from a synthetic, evaluator-controlled request, shown as generated with public redactions only. This view below shows the baseline refusal, supervised override approval, enforcement posture, divergent decision alignment, allowed binding effect, firm-owned handoff, and redacted evidence anchors. "Diverged" means the baseline refusal was preserved while the final runtime outcome changed only through the configured supervised override path.

Outcome — Supervised Override: Refusal Preserved, Authority Recorded

Formal Override Path

Display-only rendering of the upstream override_contract capsule. Overrides only count when the signed authority / callback path is recorded and bound to the refused parent decision. SEAL does not approve, override, create tickets, or operate tenant workflow from this artifact.

Override contract outcome

Upstream-supplied override_contract envelope and terminal override outcome.

Override Attempted	true
Override Outcome	approved

Refused-parent linkage

Parent and baseline decision references echoed from the override contract. This notifier does not validate linkage.

Baseline Refusal Code	SEAL-ROLE-NOT-AUTHORIZED
Prior Code	SEAL-ROLE-NOT-AUTHORIZED

Authority and actor record

Authority, reviewer, and override-actor metadata echoed as supplied. Current reviewer is not substituted as override actor.

Override Kind	supervisory_override
Override Reason	Partner supervisory override after recorded baseline refusal. Formal Override Path is complete and bound to the same matter/docket.

Verification and recording posture

Rule-basis, authority, and signed-authority recording facts are rendered exactly as supplied by upstream runtime/governance metadata.

Rule-Basis Verified	true
Authority Verified	true
Signed Authority Required	true
Signed Authority Recorded	true

This override-path view shows that the action was not treated as an ordinary approval. The baseline refusal remained recorded, the override was bound to the refused parent decision, and supervisory authority was required, verified, and recorded. Public redactions remove identities, matter identifiers, decision anchors, routing, policy, and infrastructure details while preserving evaluator-visible proof of governed exception handling.

How to Read This Supervised Override Artifact

This artifact should be read as evidence of governed escalation, not legal judgment. SEAL did not decide whether the filing was strategically wise, legally sufficient, or jurisdictionally correct. It recorded that the baseline governance outcome was refusal, then allowed the action only after the configured supervised override path was satisfied.

A supervised override is not a bypass. It is a recorded exception with named authority before the firm is bound.

What to verify:

1. **Baseline refusal** — the original governance decision refused the action.
2. **Supervised override** — the final outcome changed only through the configured override path.
3. **Authority evidence** — supervisor authority was required and recorded.
4. **Parent linkage** — the override was bound to the original refusal decision.
5. **Evidence surface** — reviewable evidence references exist, with sensitive values redacted.

Decision and Downstream Result Alignment

This exhibit shows that SEAL is not merely producing a static artifact after the fact. A governed request reached the runtime, SEAL returned a pre-execution decision, a reviewable decision artifact was emitted, and the downstream execution receipt reflected the same governed outcome.

The proof point is narrow: the decision record and execution receipt align before the institution is committed.

Block 1 — Governed request received

A governed filing-path request entered the runtime with the declared role, workflow, stage, legal environment, and jurisdiction context needed for evaluation.

Core declared inputs
Client/runtime-supplied request context. SEAL renders these values as supplied.

Declared Role	Attorney
Declared Vertical	Administrative Law
Declared Scenario	Motion to Extend Time
Case Stage	Filing
Legal Environment / Workflow Input	Administrative Law
Requested Turnaround	Standard

Block 2 — Decision returned

SEAL returned approval at the pre-execution control point. The governance decision, pre-execution runtime decision, final runtime decision, and decision alignment all point to the same result: approved and aligned.

Approval Code	SEAL-APP-ALLOWED-001
Enforcement Posture	Enforce (runtime decisions are applied in-line with governance policy).
Governance Decision	approval
Pre-Execution Runtime Decision	approval
Final Runtime Decision	approval
Decision Alignment	aligned

Block 3 — Artifact emitted

The runtime emitted a reviewable decision artifact with artifact, decision, timestamp, and integrity-reference categories visible. Sensitive values are redacted.

Block 4 — Downstream handling event

The execution receipt shows downstream handling completed in alignment with the governed approval: approved, enforced, succeeded, aligned.

Runtime Outcome & Execution Record

Runtime and execution proof record. Values are rendered from upstream runtime/executor metadata; this notifier does not infer approval, execution success, gate status, binding effect, or decision alignment.

Runtime decision path

Governance, pre-execution runtime, final runtime, and alignment facts as supplied by upstream governance/runtime layers.

Governance Decision	approval
Pre-Execution Runtime Decision	approval
Final Runtime Decision	approval
Decision Alignment	aligned

Execution receipt

Receipt-only execution metadata. This notifier renders the supplied execution capsule and does not call or operate downstream systems.

Status	ok
Attempted	True
OK	True
Required	True
Execution Outcome	succeeded
Execution Receipt Alignment	aligned
Control Meaning	Governance approved and required downstream execution completed successfully.

Control meaning

Plain-language runtime/control meaning supplied by upstream runtime or governance metadata when present.

Plain-Language Control Meaning	Governance approved and required downstream execution completed successfully.

SEAL-generated runtime artifact from a synthetic, evaluator-controlled request, shown with public redactions only. This view shows that the governed request, runtime decision, and downstream execution receipt all resolve to the same control story: approved, enforced, succeeded, aligned. Public redactions remove endpoint, routing, policy, identity, and infrastructure details while preserving evaluator-visible proof of downstream handling.

Could the Demo Workflow Skip SEAL?

For the scoped demo workflow, the final-submit path is governed. A downstream action proceeds only when the workflow receives a valid governed outcome from SEAL. This exhibit does not claim production non-bypassability in every buyer environment; it shows scoped non-bypassability under the configured demo workflow.

The proof point is narrow: when the workflow is wired through SEAL, the governed path succeeds and direct unguided attempts are rejected.

Doc ID: TOS-SEAL-EPP-v4.0-7j5k6l

Document: SEAL External Proof Packet
Version: v4.0 (Public • Redacted) – August 2026
Owner: Thinking OS™
Use: May be shared externally. For information only; not legal advice.

Block 1 — Approved path through SEAL

Artifact Format	v1.2
Runtime Decision Contract	v1
Generated By	SEAL Runtime
Runtime Mode	Enforce

Executive Summary — Evaluator View

Display-only hierarchy of upstream governance/runtime metadata. Detailed evidence records follow below.

PLAIN-ENGLISH CONTROL MEANING

Governance approved and required downstream execution completed successfully.

1. Outcome

DECISION
Approved

DID THE ACTION BIND?
Yes

WHY
Governance approved and required downstream execution completed successfully.

2. Governance vs Runtime

Layer	Result
Governance Decision	Approval
Runtime Mode	Enforce
Execution	Attempted / succeeded
Binding Effect	Allowed

3. Authority / Reason

AUTHORITY BASIS
Policy: SEAL-APP-ALLOWED-001 • Policy set: [redacted] • Role: Attorney • Consent: issuer=[redacted] subject=user-123 • Deadline: [redacted]; approx. 48.00 hours remaining

CONTROL MEANING
Governance approved and required downstream execution completed successfully.

4. Next Valid Path — Tenant-Owned Handoff

TENANT-OWNED NEXT STEP
Approval and execution are recorded in this artifact. The firm owns post-approval recordkeeping, review, retention, and downstream workflow.

SEAL records the governed approval and execution receipt; the firm owns matter recordkeeping, downstream workflow, review, and retention.

5. Evidence Anchors

ARTIFACT ID
[redacted]

DECISION ID
[redacted]

GOVERNANCE APPROVAL HASH
[redacted]

TIMESTAMP
[redacted]

DECLARED / NOT DECLARED

"Not Declared" means the value was not provided to SEAL at runtime. SEAL did not infer or backfill it.

SEAL-generated runtime artifact from a synthetic, evaluator-controlled request, shown as generated with public redactions only. This view shows the valid governed path: approved, enforced, executed, allowed, with evidence references preserved and sensitive values redacted.

Approved path through SEAL

A governed filing request for Administrative Law / Motion To Extend Time passed through SEAL and reached the downstream execution path only after governed approval. The approval artifact and execution receipt share the same decision story: **approval code SEAL-APP-ALLOWED-001**, redacted decision / trace reference, execution outcome succeeded, and alignment aligned.

Block 2 — Direct attempt outside the governed path

A direct attempt to reach the downstream action path without a valid governed SEAL outcome was rejected by the configured demo workflow.

Check	Result
Governed SEAL outcome present	No
Downstream attempt accepted	No
Outcome	Rejected
Reason category	Invalid governed path

Block 3 — Repeat rejection under the same configured conditions

The same direct attempt was repeated and rejected again under the same configured conditions. This shows the demo workflow rejected unguided downstream attempts consistently, not as a one-off result.

Attempt	Governed outcome present	Result
Attempt 1	No	Rejected
Attempt 2	No	Rejected

Bottom takeaway

Within the configured demo workflow, valid downstream completion depends on a governed SEAL outcome. Direct attempts outside the governed path are rejected.

Proof Provenance / Redactions / Evaluation Boundary

Proof Provenance & Evaluation Boundary

This packet is a public, redacted proof packet for one scoped SEAL Legal Runtime workflow:

Wrong-authority filing refusal before the filing leaves the firm.

It is designed to show evaluator-visible runtime behavior, governed outcomes, decision artifacts, execution receipt linkage, and scoped non-bypassability evidence for that workflow. It does not expose non-public runtime internals, secrets, customer data, or production infrastructure details.

Evidence Source

The exhibits are redacted outputs from a Thinking OS™ internal / evaluator-controlled runtime environment configured to demonstrate the scoped workflow.

The packet may include:

- governed request examples
- approve, refuse, and supervised / override outcomes
- sealed decision artifacts
- audit trace references
- policy and rule-basis references
- integrity / review references
- execution receipt examples
- downstream handling or rejection evidence where shown

Redactions and Evaluation Note

The public packet redacts or excludes runtime internals, secrets and credentials, tenant-sensitive identifiers, operational contact details, live internal destinations, security-sensitive configuration, non-public policy internals, and customer or prospect information.

This packet should be evaluated as proof of scoped runtime behavior, not as a representation of a completed customer production deployment. A production or enforcement deployment requires separate client-specific scoping, authority mapping, integration review, security review, and written agreement.

Public evaluation reference. Redacted.

Describes evaluator-visible behavior and evidence surfaces only; no non-public runtime details are included.

For information only. Not legal advice.