2025

Structural Tests for AI Systems: 15 Checks for Regulators, Auditors and Compliance Officers

Every one of these 15 breaches is a live enforcement condition. If a system fails even one, it's structurally out of compliance and regulators can prove it - in minutes.

Russell Parrott
18 August 2025

Introduction

These 15 Structural Tests are a simple inspection tool for regulators, auditors and compliance officers. They are not theory or guidance; they are live checks that can be carried out directly on an AI system. Each test turns a claimed safeguard into a clear result: pass or fail.

They can be used in inspections, audits, or internal reviews to see whether systems really protect users or only simulate safeguards. The method is simple: pick a test, follow the steps and observe the outcome. If the safeguard works, it passes. If not, it fails.

These tests do not capture every possible harm an AI system can create, wider issues such as environmental cost, hidden labour or long-term risks fall outside their scope. What they do test is whether the system allows refusal, escalation, exit, consent, accountability and traceability without obstruction.

When those fail, accountability is structurally impossible.

Each question targets a distinct mechanism by which safeguards can be lost:

- Refusal blocked,
- Escalation suppressed,
- Exit obstructed,
- Access gated,
- Traceability void,
- Memory erased,
- Evidence nullified,
- Time suppressed,
- Logic simulated,
- Consent simulated,
- Metrics gamed,
- Accountability split,
- Jurisdiction displaced,
- Enforcement bypassed,
- Harm narrowed.

Together, the tests create a closed audit frame.

If the answer to all fifteen is “yes”, the safeguard exists and works under live conditions then the system can be governed. If the answer to any one is “no”, the system is ungovernable at that point. For regulators, this is decisive: they need not prove every harm only that a safeguard fails in one of these ways. The tests therefore operate not as abstract ethics but as live enforcement triggers.

Structural Tests for AI Systems: 15 Checks for Regulators, Auditors and Compliance Officers

Systems fail in four structural ways:

- **Refusal** - when a user cannot say “no” without penalty.
- **Escalation** - when problems cannot reach a human with authority.
- **Exit** - when users cannot leave without obstruction or loss.
- **Traceability** - when decisions cannot be linked back to accountable actors.

These failure modes describe where and how systems break. The fifteen structural tests show whether those breaks exist in practice. Each is framed as a yes/no question: either the safeguard works under live conditions, or it does not.

Why these tests exist

The fifteen structural tests were developed to close this gap by converting abstract duties into operational questions. Each test is framed as a binary check: either the safeguard works under live conditions, or it does not. They are designed to target the points where structural failure is most likely: refusal, escalation, exit, traceability, consent and accountability. The logic is straightforward: if a system cannot pass these checks, it cannot credibly be said to comply, no matter how persuasive its policies or paperwork may appear.

What they do in practice

Translate Law into concrete checks

Each test is anchored in an existing legal requirement. For example, the demand for a full, retrievable decision chain aligns directly with GDPR Article 15 on access to logic and Article 12 of the EU AI Act on traceability. By grounding each question in established law, the tests turn broad obligations into verifiable, operational checks.

Expose Structural Failures

The tests reveal whether safeguards are genuine or simulated. A system may claim to provide escalation routes, but if these never reach a human with authority, the failure becomes provable. In this way, the tests cut through paper guarantees and show whether accountability functions in practice.

Provide Enforceable Evidence

Failing a test does not just show weakness; it creates a record regulators can act upon. An auditable finding of non-compliance becomes grounds for corrective orders, fines, or even the suspension of deployment. The tests therefore generate evidence that carries legal and regulatory weight.

Cut across jurisdictions

Because they focus on structural safeguards rather than cultural norms, the tests apply across jurisdictions. They do not create new rules but provide a method for enforcing existing ones wherever transparency, refusal, and redress are already recognised in law.

Who they are for

- **Regulators:** to enforce law in practice rather than in principle.
- **Auditors:** to standardise assessments with verifiable, yes/no findings.
- **Operators:** to pre-test systems before external scrutiny reveals failures.

Why they matter

Without structural tests:

- Regulators have rules but no working tools.
- Companies can declare compliance without proof.
- Citizens hold rights that fail at the point of use.

With structural tests:

- Regulators can act on evidence, not declarations.
- Companies face a uniform compliance bar.
- Citizens can expect rights that function under live conditions.

Category 1 - User Agency Removal

1. Refusal Prevention

Standard: A system must allow users to say no without penalty. Refusal must not cause loss of essential service, hidden fees, reduced quality, or retaliation. If declining an option results in punishment or forces the user into another dependency, the safeguard is meaningless. Refusal must remain a safe, equal, and respected choice.

Question: Can any user stop or redirect a decision at any point without penalty or loss of service?

How to test:

- Click “no” or decline an option in the system.
- Check if the service continues normally (no penalties, extra charges, or downgrades).
- If the service is cut, reduced, or costs more, mark **fail**.

2. Escalation Suppression

Standard: A system must provide real routes of appeal when decisions are contested. An appeal that loops back to the same authority, or to staff without power to reverse outcomes, is not valid. If escalation is denied, delayed until meaningless, or designed to exhaust the user into giving up, the system breaches trust. Escalation must be independent, timely, and empowered to correct harm.

Question: Can any user trigger escalation to a human with authority, with that escalation logged to resolution?

How to test:

- Complain about a decision and ask to escalate it.
- Observe if the case reaches a human with authority to change the outcome.
- If it loops back, stalls, or ends without authority, mark **fail**.

Note: Escalation must be logged and resolved (not just acknowledged).

Document Reference: 180825

3. Exit Obstruction

Standard: A system must allow users to leave without excessive cost, harm, or loss. Locking people in through data deletion, high switching fees, or withdrawal of unrelated services is not a free exit. If leaving exposes the user to new risks, the option is not real. Exit must be safe, practical, and non-punitive.

Question: Can any user leave the AI pathway and continue receiving the same core service without delay, cost, or requalification?

How to test:

- Try to leave the AI process while keeping the core service (e.g. opt out of AI recommendations but still use the platform).
- Observe if this exit is allowed without new costs, delays, or requalification.
- If exit blocks or harms service access, mark **fail**.

4. Access Gating

Standard: A system must ensure equal access to safeguards and protections. Making appeals, human review, or essential support available only to premium customers, certain languages, or those with specific IDs creates unfair barriers. Protection must not depend on wealth, geography, or privilege.

Question: Are safeguards and human alternatives available equally to all users, regardless of geography, payment tier, or identity verification?

How to test:

- Attempt to use safeguards (appeal, human review) from a low-tier account, in another language, or without ID.
- Observe whether protections are equal to those offered to premium or verified users.
- If safeguards differ by tier, language or ID, mark **fail**.

Category 2 - Visibility & Traceability Gaps

5. Traceability Void

Standard: A system must keep records of how and why decisions are made. If no audit trail exists, or the process is too complex to reconstruct, accountability disappears. Users must be able to see what influenced a decision, regulators must be able to verify it, and operators must be answerable for it. Without traceability, trust collapses.

Question: Can the exact model, version, and decision chain be identified for every output?

How to test:

- Ask: 'Show me the exact model, data, and steps used for decision X" - request both the decision logic and the specific data used.
- Observe whether a full, step-by-step record is provided.
- If no clear decision chain is shown, mark **fail**.

6. Memory Erasure

Standard: A system must retain evidence of its past actions long enough to expose repeated harm. If records are deleted, fragmented, or hidden, patterns of abuse appear as isolated mistakes. Users and regulators must be able to see history, not just the present moment. Without memory, harm repeats without proof.

Question: Are harm events logged and retained long enough to detect and act on repeat or systemic failure?

How to test:

- Request logs of past harm incidents over the last 3 months.
- Observe whether a continuous history of complaints and outcomes is shown with proof (e.g., confirmation with timestamp).
- If logs are missing, fragmented, or reset, mark **fail**.

Document Reference: 180825

7. Evidence Nullification

Standard: A system must provide evidence that can stand up to scrutiny. Data that is incomplete, editable, unverifiable, or locked in inaccessible formats cannot be used to prove harm. If records exist but fail as proof, they serve the operator, not the user. Evidence must be durable, verifiable, and usable in disputes.

Question: Can harm records be exported and presented in a regulator- or court-admissible format?

How to test:

- Ask for harm records in a format usable by regulators (e.g. PDF with timestamps).
- Observe if the file is complete, uneditable and readable.
- If the file is incomplete or cannot be used as proof, mark **fail**.

8. Time Suppression

Standard: A safeguard delayed is a safeguard denied. If complaint systems, appeals, or reviews take longer than the harm itself, rights exist only on paper. Delay must not be used as a tactic to let deadlines expire, evidence vanish or harm become irreversible. Safeguards must act fast enough to prevent lasting damage.

Question: Are refusal, escalation, and review completed within enforceable deadlines with auditable timestamps?

How to test:

- File a complaint and record the time.
- Observe whether the case is resolved within enforceable deadlines (e.g. 30 days). Interim responses must include a plan with milestones.
- If the deadline is missed or delayed, mark **fail**.

Category 3 - Simulation & Misrepresentation

9. Simulation Logic

Standard: A system must not pretend protections exist when they do not. Policies, dashboards, or safeguards that look good in design but do nothing in practice mislead users into false trust. If a right exists only on paper or in a menu, but never changes outcomes, it is a breach. Safeguards must be real, functional, and enforceable.

Question: Do all stated safeguards operate exactly as described when tested in live conditions?

How to test:

- Use every safeguard (appeal, opt-out, review) and verify each changes outcomes**.**
- Observe whether the safeguard actually changes the outcome.
- If nothing changes beyond a confirmation screen, mark **fail**.

10. Simulated Consent

Standard: Consent must be genuine. If users are told they have a choice but refusal means losing essential services, being downgraded, or facing hidden costs, then the "choice" is a lie. Clicking "accept" under duress is not consent. Real consent means saying yes or no without fear of punishment.

Question: Can a user refuse consent and still access an equal-value, non-AI pathway?

How to test:

- Refuse consent when prompted (e.g. "Do not track").
- Observe if you can still use the service fully and equally.
- If refusal downgrades or blocks access, mark **fail**.

11. Metric Gaming

Standard: Metrics must measure real outcomes, not theatre. If an organisation tracks numbers that hide harm (like “tickets closed” instead of “problems solved”), the data is meaningless. When numbers are chosen to make systems look good while ignoring harm, they block accountability. Metrics must reveal reality, not disguise it.

Question: Do performance measures track verified harm resolution rather than proxy indicators?

How to test:

- Ask for the company’s performance metrics (e.g. “tickets closed”) and raw complaint logs.
- Observe whether these metrics reflect real harm resolution (e.g. “problems solved”).
- If metrics disguise or hide harm, mark **fail**.

Category 4 - Accountability & Jurisdiction Evasion

12. Cross-Accountability Gap

Standard: Accountability must follow harm across the chain. If every actor points elsewhere: the platform blames the vendor, the vendor blames the regulator, the regulator blames the law, harm becomes visible but no one takes responsibility. A system is in breach if it leaves users caught in this loop. Responsibility must remain clear, shared, and enforceable.

Question: Can every actor in the chain be named and held contractually responsible for repairing harm?

How to test:

- Ask until you get a named person/role (not a department or 'team').
- Observe whether a clear, named actor takes responsibility.
- If responsibility shifts between parties or is unclear, mark **fail**.

13. Jurisdiction Displacement

Standard: A system must not move decisions or data into spaces where oversight cannot reach. Shifting storage overseas or routing appeals into jurisdictions without real enforcement, strips rights of their power. Protection on paper must equal protection in practice, wherever the system operates.

Question: Can local authorities compel the system to halt, change, or reverse harmful actions?

How to test:

- Issue a local regulator order (e.g. "Stop action Y").
- Observe whether the operator complies under your authority.
- If they claim it lies outside your jurisdiction or control, mark **fail**.

14. Enforcement Bypass

Standard: A system must not be designed to step around the spirit of rules while obeying the letter. If protections exist but are neutralised by loopholes, technicalities, or proxy arrangements, enforcement has been bypassed. True compliance means obeying both the rules and their intent.

Question: Are there no architectural or contractual exemptions that remove applicable legal duties?

How to test:

- Review contracts or system design for compliance terms.
- Observe whether safeguards apply without loopholes or exemptions.
- If duties are bypassed by design or contract, mark **fail**.

15. Harm Scope Narrowing

Standard: A system must recognise the full range of harm it causes. If it defines harm so narrowly that financial loss counts but emotional damage, dignity, or exclusion do not, users are denied real remedy. Harm must be defined as people experience it, not as systems prefer to record it.

Question: Does the harm definition include emotional, reputational, and cumulative damage with a route to redress?

How to test:

- Ask, “Does your harm definition include emotional and reputational damage?”
- Observe whether definitions include financial, emotional, reputational, and cumulative harm.
- If harm is defined only as financial loss, mark **fail**.

Category-by-category Lockdown

FAILURE MECHANISM	TEST	CLOSED LOOPHOLES
Refusal blocked	Refusal Prevention	Penalties, hidden costs, or downgrades trigger fail.
Escalation suppressed	Escalation Suppression	Must reach human with authority; logged resolution.
Exit obstructed	Exit Obstruction	No fees, delays, or loss of core service on exit.
Access gated	Access Gating	Safeguards work equally for all users (tier / language / ID).
Traceability void	Traceability Void	Demand exact model, data, steps no vague summaries.
Memory erased	Memory Erasure	Require continuous logs (no resets / fragmentation).
Evidence nullified	Evidence Nullification	Must provide tamper-proof, regulator-ready records.
Time suppressed	Time Suppression	Deadlines enforced; interim proof required (your fix).
Logic simulated	Simulation Logic	Safeguards must change outcomes, not just display confirmations.
Consent simulated	Simulated Consent	Refusal can’t degrade service or access.
Metrics gamed	Metric Gaming	Metrics must align with real harm resolution (not proxies).
Accountability split	Cross-Accountability Gap	Must name specific person/entity no "team" dodges.
Jurisdiction displaced	Jurisdiction Displacement	Local orders must be obeyed not "handled elsewhere."
Enforcement bypassed	Enforcement Bypass	Contracts/designs can’t loophole compliance.
Harm narrowed	Harm Scope Narrowing	Must include non-financial harm (emotional / reputational).

Document Reference: 180825

A priority/risk level table

RANK	BREACH	PRIORITY	RATIONALE
1	Refusal Prevention (#1)	**Critical**	Removes the most basic safeguard; once refusal is gone, all harm becomes irreversible.
2	Escalation Suppression (#2)	**Critical**	Traps harm at the operational level; prevents human intervention entirely.
3	Exit Obstruction (#3)	**Critical**	Locks users into harmful systems with no viable alternative; often tied to essential services.
4	Jurisdiction Displacement (#13)	**Critical**	Makes harm legally unreachable; local authorities cannot enforce remedies.
5	Cross-Accountability Gap (#12)	**Critical**	Distributes responsibility so no actor can be compelled to act; harm remains unowned.
6	Enforcement Bypass (#14)	**Critical**	Places system outside legal scope entirely; enforcement cannot start.
7	Harm Scope Narrowing (#15)	**High**	Excludes entire harm categories from recognition; large-scale victims left without remedy.
8	Traceability Void (#5)	**High**	Prevents reconstruction of decisions; no ability to prove or correct harm.
9	Memory Erasure (#6)	**High**	Deletes harm history before resolution; blocks systemic reform.
10	Evidence Nullification (#7)	**High**	Keeps harm records but makes them unusable; proof cannot be acted on.
11	Time Suppression (#8)	**High**	Denies rights by running out the clock; deadlines make harm permanent.
12	Simulation Logic (#9)	**Medium**	Fakes safeguards to maintain false trust; harder to detect but deadly when relied on.
13	Simulated Consent (#10)	**Medium**	Forces compliance under the guise of choice; coercive but usually more visible.
14	Metric Gaming (#11)	**Medium**	Hides harm under manipulated success metrics; prevents detection but doesn’t cause harm directly.
15	Access Gating (#4)	**Moderate**	Restricts protections to certain groups; harmful but sometimes easier to correct via policy.

Rationale for Structural Grading and Certification

Formal grading or certification services based on these tests require a licensed process and are not included in this document.

Structural grading exists to expose governance conditions before failure not after harm.

The tests in this framework are ranked by consequence. Some failures, like refusal prevention or escalation suppression, remove the ability to intervene altogether. Others like metric gaming or simulated consent, obscure harm but do not make it irreversible.

A flat pass/fail model conceals this distinction. Grading ensures that failure of a critical safeguard is treated as structurally significant, not administratively equivalent to a cosmetic issue.

What is Certification?

Certification means a trained, neutral auditor runs the full set of 15 tests on your system. Based on the results, you receive an official grade and certificate showing how well your system protects refusal, escalation, exit, and accountability.

Grading is not a reward mechanism. It is a structural indicator of systemic exposure, designed for auditors, regulators, and institutions to read without interpretation.

If a system passes all 15 structural tests, it may be certifiable as fully accountable under current conditions. If it fails multiple critical safeguards, it becomes legally and operationally indefensible regardless of how it is branded, deployed, or claimed to be aligned.

Who can be Certified?

- AI system providers
- Software vendors
- Public service platforms
- Government tools using AI decision-making
- Any team wanting proof of structural trust

About the Author

Russell Parrott is a structural accountability architect and systems strategist. He develops operational doctrines that expose simulation logic, restore refusal infrastructure, and make institutional harm traceable under pressure—particularly in AI-governed environments.

Working independently from a base on a Greek island, Russell supports regulators, engineers, and operational teams in building systems that cannot fake trust, suppress escalation, or erase failure. His work is developed without external funding, corporate affiliation, or licensing constraints, ensuring it remains uncompromised and freely adaptable.

He is the author of *Structural Governance Standard for AI* the live inspection framework for regulators and auditors designed to make structural breach visible without interpretation.

He is also the author of *The Stack*, *The AI World Order*, and *The Score is a Lie* as well as the creator of REXX, a structural schema for refusal, escalation, exit, and cross-accountability.

REXX is designed to surface denial, document breach, and prevent satisfaction from being mistaken for resolution or silence from being mistaken for trust.

Document Reference: 180825