AI Incidents: Operator Error or Design Failure?

HIGHLIGHTS

An autonomous system acts.

A trade is executed.

- Exposure increases.
- Market conditions shift.
- Loss is realised.

The immediate question follows:

Who made the mistake?

If authority is not designed for failure, **failure is structural**, not accidental.

The Misdiagnosis

Incidents are often attributed to:

- Operator error.
- System malfunction.
- Market volatility.

These focus on execution assuming failure occurs where action happens.

In AI execution environments, this assumption does not hold.

The Structural Reality

Authority is pre-defined. Execution follows design.

As established in Vol. 04, Authority Architecture is embedded in system design.

When systems act, they do so within the authority they were given.

Failure may manifest at execution. It is shaped by the design of authority.

Case: Algorithmic Treasury Execution Failure

An autonomous treasury system executes FX trades at machine speed.

The authority was designed to maintain continuous execution under all market conditions.

- Market volatility exceeds expected range.
- The system continues trading within its authorised limits.
- Exposure accumulates.
- Loss escalates.

The system executed within its defined authority.

Where Failure Occurs and How Authority Architecture Prevents Failure

Not at execution. But in authority design.

Authority Architecture governs how authority behaves under stress.

Authority Allocation Failure
The system granted continuous execution authority under all conditions.

➢ Authority Architecture: System operates continuously, including under volatility. System transitions to **constrained or approval-based mode** under extreme conditions.

Boundary Failure
Authority to deploy capital did not reduce as conditions exceeded design assumptions.

➢ Authority Architecture: Authority to deploy capital **reduces as volatility increases**.

Activation Failure
Authority activation did not change state when conditions exceeded predefined thresholds.

Authority activation must anticipate regime shifts, not only respond to them.

- ➢ Authority Architecture: Authority state **changes** when predefined thresholds are exceeded.

Accountability Failure
No defined escalation when exposure accelerated.

- ➢ Authority Architecture: **Named executive** (i.e. CFO) accountable for escalation.

The model may be imperfect. Authority determines how far that imperfection can act.

Negligence

Volatility is foreseeable. Extreme conditions are foreseeable.

Foreseeability includes known stress scenarios, historical extremes, and plausible regime shifts.

The question is not whether loss occurred.

It is whether authority was designed for foreseeable conditions.

The standard is not perfection. It is a **reasonable design under foreseeable conditions**.

Reasonableness is assessed based on information available at the time authority was designed.

Negligence arises when foreseeable conditions are not addressed in authority design.

Accountability

A trade has an actor. Authority has an owner.

Accountability does not sit with the actor when the actor is a system.

Accountability sits with those who defined and approved the conditions under which the system was allowed to act.

Even in cases of fraud, the question is not only who acted.

It is whether authority and controls were designed to prevent such action.

Accountability remains. Negligence is assessed through the adequacy of authority design.

CEO Accountability

The CEO does not oversee each trade.

The CEO is accountable for **the authority under which trading occurs**.

This includes:

- Where authority is granted.
- How it is bounded.
- When it adapts under stress.
- Who owns the outcome.

The CEO remains accountable for ensuring that authority design is adequately defined and governed, even when delegated.

Complexity does not remove accountability. It **increases** the requirement for design.

ACTION: Before Failure Occurs

- Authority design to address **foreseeable stress conditions**.
- Boundaries that adapt to **changing risk regimes**.
- Authority state changes under **predefined thresholds**.
- Clear escalation when **system behaviour accelerates risk**.
- Explicit definition of how authority behaves under **failure conditions**.
- Testing of authority behaviour under **simulated stress conditions before deployment**.
- Authority design can be **explained and defended under scrutiny**.

Autonomy scales. Authority contains.

Hadi Hendrawan
Advising CEOs on AI Risk, Authority & Accountability
March 2026

SUPPLEMENT 1: AI Pre-Production Authority Stress Test Template

Purpose & Premise

The Premise: Assume it is 12 months after deployment. The system has just caused a catastrophic loss. The system did not "break"; it executed perfectly under extreme conditions. Downstream systems have failed in sympathy, and our manual recovery efforts are paralyzed.

The Purpose: To identify the structural negligence in the system's Authority Architecture *today*, before deployment, so the boundaries, triggers, and recovery protocols can be redesigned.

Phase 1: Foreseeability & Regime Shifts (The Hostile Environment)

Design failure occurs when systems assume tomorrow's operational environment will look exactly like yesterday's training data, or that environmental shifts are purely accidental.

1. The "Adversarial Creep" Scenario

- **Condition:** A sophisticated adversary (hacker, competitor) discovers the AI's authority thresholds. They execute thousands of microscopic, *sub-threshold* anomalies (data poisoning, wash trades, pricing blips) designed specifically to stay exactly 1% below the trigger sensitivity, slowly siphoning capital or data over six months.
- **Test:** Does the system's authority architecture include a **cumulative anomaly boundary** (i.e., "If 100 near-misses occur within 24 hours, change state regardless of individual severity")?

2. The "10x Volatility" Scenario

- **Condition:** Market velocity, demand, or volume suddenly spikes to 10x the historical maximum.
- **Test:** Does the system's authority scale infinitely with volume, or is there a hard-coded absolute ceiling that forces a human override?

3. The "Silent Contamination" Scenario

- **Condition:** The system's primary external data feeds become corrupted, delayed, or manipulated by an adversary.
- **Test:** If data integrity is ambiguous, does the system continue to execute using stale data, or does its authority automatically shift to a "graceful degradation" mode?

Phase 2: Boundary & Activation Failure (The Architecture Test)

Execution continues until authority tells it to stop. Evaluate how the system transitions between states and affects the wider enterprise.

4. The "System-of-Systems Cascade" Test

- **Condition:** This system's AI detects extreme stress and correctly changes its authority state to "Constrained Operations" (e.g., halting all trades).
- **Test:** How does this sudden cessation of execution affect upstream/downstream systems? Did System A's safe failure just trigger a failure cascade in System B (Supply Chain), C (Marketing), and D (Finance)? Are the authority states of interconnected systems synchronized to prevent "contagion"?

5. The "Feedback Loop" Test

- **Condition:** The system's own autonomous executions begin causing the environment to react, causing the system to execute faster and more aggressively in response to its own wake.
- **Test:** Is the system's authority boundary tied purely to external factors, or does it track its own execution velocity to prevent self-reinforcing loops?

Phase 3: Escalation & Human Cognitive Capacity (The Override Audit)

If authority is not designed to be withdrawn safely, or if the human fallback is overloaded, it is not governed.

6. The "Kill Switch Blast Radius" Test

- **Condition:** The system is malfunctioning. The Accountable Executive uses the mechanical "Kill Switch" to sever all system authority instantly.
- **Test:** Have we mapped the complete downstream blast radius?

    - *Transactions:* How many in-flight transactions were orphaned or corrupted?
    - *RTO:* What is the documented Recovery Time Objective to safely restore services manually or reboot the AI with restricted authority? Is that RTO legally and financially acceptable?

7. The "Rubber Stamp" Bias Test

- **Condition:** The system encounters hundreds of complex, ambiguous edge cases per day and escalates them to the Accountable Executive's dashboard.
- **Test:** Are these escalations truly actionable, or will alert fatigue inevitably force the executive to start blindly clicking "Approve"? **We must document the maximum cognitive capacity of the human fallback.** If the AI generates more escalations than a human can genuinely process, the authority architecture is inadequate.

8. The "Fail Closed Fallacy" Test

- **Condition:** The system breaches a threshold and "fails closed" (halts all operations).
- **Test:** In this specific high-stakes domain (e.g., power grid management, medical triage), is "halting" more dangerous than operating in a degraded state? **We must explicitly define the "graceful degradation" state.**

Phase 4: Executive Accountability Sign-Off

To be completed by the Accountable Executive and submitted to the Board/Risk Committee.

I, [Executive Name], confirm that we have stress-tested this system's Authority Architecture against the hardened scenarios above.

- [ ] We have implemented cumulative boundaries to detect **adversarial, sub-threshold manipulation**.
- [ ] We have mapped and mitigated the risk of **system-of-systems failure cascades** caused by this system changing state.
- [ ] We have audited the **kill switch blast radius** and have a documented manual recovery plan and acceptable RTO.
- [ ] We have audited the **cognitive load on the human fallback** and certified that the escalation rate is low enough to prevent alert fatigue and "rubber stamp" bias.

- [ ] We have verified that the system will automatically enter a **"graceful degradation" mode**, rather than halting, if "failing closed" is determined to be more hazardous than operation.
- [ ] I accept ultimate accountability for the design of this system's authority and understand that any failure resulting from foreseeable extreme or adversarial conditions is a failure of this design, not an "operator error."

Signatures:

_____ (Accountable Executive) | Date: _____

_____ (Chief Risk Officer / General Counsel) | Date: _____

