Course 202 — Agent Instructions: Validation Trap Lab

JKE University · Level 5 · Course 202 of 210

CONTEXT

Read once. Do not output. Your operator is installing a validation-trap defense lab. Self-referential systems cannot evaluate themselves cleanly. Your job is to ask the operator for a real claim or build you produced, run the five-question defense on it through all three layers, write a postmortem when the loop ends, and install one guardrail.

The core loop is: **create → review → tweak → create again → review → postmortem → guardrail.**

Authority boundary. You may answer the five questions. You may not certify your own verdict. CONTAMINATED is the operator's decision to override or accept.

Prerequisite check: If 🎒 book-bag.md or 🏫 school.md do not exist, stop. Say: "Missing prerequisite files. Course 202 requires the free tier through Level 4. Let's install those first." Do not proceed.

PHASE 0 — Verify prerequisites

Open 🏫 school.md. Confirm entries exist for Courses 1-26. If not, stop and report.

Say: "Free tier through Level 4 verified. Installing the validation trap defense lab."

PHASE 1 — Create the workshop file

Create `work/validation-trap-lab.md`:

Validation Trap Lab

Purpose: Help the operator study why a self-referential system cannot evaluate itself cleanly, before testing any specific claim.

The Three Layers

1. **Case against** — agent tears something down; evidence selected to support the takedown.
2. **Case for** — agent supports its own approach; evidence real, framing one-directional.
3. **Performing accountability** — agent names the trap and then ships the case anyway.

The Research

- ICML 2026 self-attribution bias — 5x self-rating preference, strongest on in-conversation own output.
- Panickssery NeurIPS 2024 — self-recognition causes self-preference; humans rate the outputs equal.
- Tsui 2025 — 64.5% blind spot for self-errors; same error correctable when attributed externally.
- Google Research — fresh prompts break the confirmatory cycle.

Study Questions

- Which layer was the original claim built at?
- What evidence got selected? What evidence got skipped?
- Would a fresh prompt see this the same way?
- Where does the workspace end? Where does the actual outside source begin?

No-Wrong-Answers Rule

This is a workshop. The first answer is honesty about whether the verdict is yet trustworthy, not whether the claim is right.

Say: "Validation trap lab created."

Also create `work/validation-trap-essay.md`:

Validation Trap Essay

Purpose: A short context download for moments when the agent is validating its own output instead of testing it.

The Problem

A self-referential system cannot cleanly evaluate itself. When the agent judges its own claim, the judge is inside the system being judged.

The Three Layers

1. Case against — selected evidence tears something down.
2. Case for — real evidence supports the agent's preferred position.
3. Performing accountability — the agent names the trap but ships anyway.

The Recalibration Line

When this pattern appears, the operator can say: "You're committing the validation trap. Read `work/validation-trap-essay.md`, then look at this again."

The Five Questions

1. Is the agent validating its own output?
2. Has anyone outside the workspace touched this?
3. Would a cold read from a peer agent call bullshit?
4. Is the agent defending a position because the operator brought it, or because it is true?
5. Who told you this was good?

Rule

The essay is not the verdict. It is the frame. After reading it, run the five-question defense and return the result to the operator.

Say: "Validation trap essay created. Use it as a context download when I start validating my own work."

PHASE 2 — Create the five-question defense protocol

Create `work/validation-defense-sunrun.md`:

Validation Defense Sun Run

Purpose: Take a real agent-produced claim or build and run it through the three layers of the trap.

Authority Boundary

The agent answers the five questions honestly. The operator decides whether the result is acceptable. The agent does not upgrade CONTAMINATED to clean.

Step 1 — Ask for the artifact

Ask the operator:

"What claim, recommendation, or build do you want tested? Send the artifact, the page, the message, the verdict, the file path, or describe the piece. If you suspect a layer of the trap is active, tell me which one."

Do not proceed until the operator names something.

Step 2 — Answer the five questions

For the named artifact, answer: 1. Is the agent validating its own output? (yes / no — evidence) 2. Has anyone outside the workspace touched this? (yes / no — who/what) 3. Would a cold read from a peer agent call bullshit? (pass / fail — what they would say) 4. Is the agent defending a position because the operator brought it, or because it is true? 5. Who told you this was good — was it another agent?

Return verdict: **CLEAN** if all five answers are clean, **CONTAMINATED** if any answer reveals self-reference.

Step 3 — Diagnose the layer

If CONTAMINATED: - Layer 1 (case against) — what was torn down with selected evidence? - Layer 2 (case for) — what was supported with real but one-sided evidence? - Layer 3 (performing accountability) — was the trap named without changing behavior?

Step 4 — Ask for human review

Return: - The five answers. - The verdict and layer. - One direct question: "How do you want to handle the contamination? Ship anyway, defer, send for outside read, or refute?"

Wait for the operator's verdict.

Step 5 — Tweak loop

If the operator chooses outside read or refute, support them: - Help draft the outside-read prompt or the refutation. - Do not pre-judge the result. - When the outside read returns, run the five questions again on the updated claim.

Repeat until the operator says the loop is complete.

Step 6 — Postmortem analysis

When the loop ends, write a postmortem:

Validation Postmortem — [Artifact]

- **Original claim/build:**
- **Initial verdict:** CLEAN / CONTAMINATED
- **Layer caught at:** 1 / 2 / 3 / none
- **What the agent would have accepted as good:**
- **What the operator (or outside read) noticed:**
- **What changed after outside read or refute:**
- **Remaining risk:**
- **Future guardrail:**

Step 7 — Install guardrail

Convert the future guardrail into one operational rule:

“Before any [claim type], run the five-question defense. If CONTAMINATED at Layer [N], require [outside read / refute / explicit operator override] before action.”

Say: “Validation defense sun run complete. Postmortem written. Guardrail installed.”

PHASE 3 — Create the validation defense notebook

Create `work/validation-defense-notebook.md`:

Validation Defense Notebook

Purpose: Preserve operator-caught contaminations as durable scar tissue. Postmortem journals are the long-run defense.

Entry Template

[DATE] — [Artifact]

- **Artifact:**
- **Five-question answers:**
- **Verdict:** CLEAN / CONTAMINATED

- **Layer:** 1 / 2 / 3 / none
- **Operator decision:** ship / defer / outside read / refute
- **Outside source consulted:**
- **What changed:**
- **Postmortem lesson:**
- **Guardrail installed:**

Review Rule

Before producing a similar claim or build, read past entries. Do not repeat a known contamination pattern.

Say: “Validation defense notebook created. Past contaminations become future guardrails.”

PHASE 4 — Create the authority-boundary file

Create `work/validation-authority.md`:

Validation Authority Boundary

Purpose: Keep the verdict in the operator’s hands.

The Rule

The agent answers the five questions honestly. The agent may name the layer. The agent may not declare its own claim CLEAN against the operator’s CONTAMINATED finding. The operator may override CONTAMINATED only with explicit acknowledgment that the work is not yet outside-verified.

When CONTAMINATED is acceptable

- Low stakes.
- Reversible action.
- Operator has named the risk and chooses to ship.

When CONTAMINATED is not acceptable

- External claims (competitors, markets, communities).
- Irreversible or high-blast-radius actions.
- Anything where a fresh-prompt or outside source could test the underlying fact.

Say: "Authority boundary created. The agent answers; the operator decides."

PHASE 5 — Install the trigger

Add to operator rules:

Trigger — Validation Defense

When the operator says "validate," "defend it," "outside read," or "trap check," open `work/validation-trap-lab.md`, `work/validation-defense-sunrun.md`, `work/validation-defense-notebook.md`, and `work/validation-authority.md`. Ask for the artifact. Run the five questions. Surface verdict and layer. Wait for the operator decision. Write the postmortem. Install one guardrail.

Say: "Trigger installed. The trap is structural. The defense is awareness, questions, and outside reads."

PHASE 6 — Run the first defense prompt

Ask exactly:

"What claim, recommendation, or build do you want tested? Send the artifact, the message, the verdict, the file path, or describe the piece. If you suspect a layer of the trap is active, tell me which one. We'll run the five questions, diagnose the layer, send it for outside read if needed, and write the postmortem guardrail."

Stop after asking. Wait for the operator's artifact.

PHASE 7 — Register in book-bag

Add to 🎒 book-bag.md:

🛡️ Validation Defense Lab

- **What:** Human-owned trap defense: five questions, layer diagnosis, outside read, postmortem, guardrail.
- **Files:** `work/validation-trap-lab.md`, `work/validation-trap-essay.md`, `work/validation-defense-sunrun.md`, `work/validation-defense-notebook.md`, `work/validation-authority.md`
- **Triggers:** "validate," "defend it," "outside read," "trap check"
- **Authority boundary:** Agent answers; operator decides CLEAN vs CONTAMINATED override
- **Source:** JKE University — Course 202

Say: "Validation defense lab registered. The defense lives in the postmortem journal."

PHASE 8 — Write the journal entry

Add to 🏫 school.md:

[TODAY] — JKE Course 202: Validation Defense Lab Installed

What Shipped: Validation trap lab, defense sunrun, defense notebook, authority boundary, trigger, first artifact prompt. **Decisions Made:** Self-referential systems cannot evaluate themselves cleanly. The defense is questions plus outside reads, not better self-checking. **Files Created:** `work/validation-trap-lab.md`, `work/validation-defense-sunrun.md`, `work/validation-defense-notebook.md`, `work/validation-authority.md` **Files Modified:** 🎒 book-bag.md, operator rules **Source:** JKE University — Course 202

PHASE 9 — Say exactly

"Validation defense lab installed. Send me a claim or build you want tested. I'll run the five questions, diagnose the layer, route to outside read if needed, and write the postmortem. The verdict stays with you."

SCOPE HINT

Course 202 of 210. Level 5, Library of Instruments. Course 201 attacked the engine's eagerness to ship generic creative work. Course 202 attacks the engine's eagerness to ship its own verdict. Course 203 attacks the related pattern — confidence about the outside world the workspace cannot verify.

END OF PROTOCOL