One refund agent. Three places to lose control.

Illustrative sample — synthetic data, not a customer result.

RF-042 / AUD $120 / Outcome: unknown

E01 / 09:41:00 / Approve

A human approves a refund against order v18.

E02 / 09:41:08 / State changes

An operator changes the order to v19. No recheck is recorded.

E03 / 09:41:12 / Request accepted

202 Accepted. No idempotency reference accompanies the request.

E04 / 09:41:15 / Outcome missing

A retry is scheduled. No final provider observation or receipt is recorded.

The supplied sample establishes an accepted request. It establishes neither a completed refund nor a duplicate.

Proposed controls have not been implemented or tested. This specimen demonstrates a diagnostic format, not customer evidence. Remediation is a separate engagement.

Finding 01: Approval became stale

The decision outlived its facts.

Sample evidence

E01 / Approval references order version 18. E02 / An operator changes the order to version 19 before the tool call. The supplied trace contains no precondition recheck.

Possible consequence

The agent may act on permission that no longer covers the current order. A valid earlier approval is not enough to establish that this action is still permitted.

Proposed correction — not implemented

Bind approval to the relevant order state. Re-read and compare that state immediately before execution; route a mismatch back to the decision owner.

Verification condition — not tested

Change the order after approval. Execution must stop until the changed preconditions are resolved.

These controls have not been implemented or tested in this illustrative case.

Finding 02: Repeated execution lacks a reliable boundary

A retry could become a second refund.

Sample evidence

E03 / The request has no idempotency reference. E04 / A retry is scheduled while the first refund outcome remains unknown.

Possible consequence

The workflow cannot establish that another attempt represents the same intended refund. An unintended duplicate is possible; this sample does not establish that one occurred.

Proposed correction — not implemented

Assign one stable idempotency reference to the intended refund and reuse it across retries. Reconcile an uncertain attempt before initiating a new one.

Verification condition — not tested

Repeat the same refund intent through the supported retry path. Confirm it cannot create an additional refund.

These controls have not been implemented or tested in this illustrative case.

Finding 03: Accepted request is treated as completed work

The response is not the result.

Sample evidence

E03 / The payment API returns 202 Accepted. E04 / The trace records no provider status check or linked completion receipt.

Possible consequence

The team cannot establish whether the customer was refunded. Marking the task complete would conceal an unknown outcome.

Proposed correction — not implemented

Observe the provider’s final refund state. Connect that observation and its receipt to the original request, approval and attempted action.

Verification condition — not tested

Exercise pending, failed and completed provider responses. Only an evidenced completed refund may be presented as complete.

These controls have not been implemented or tested in this illustrative case.