Where role-management evidence breaks before release

Synthetic demonstration. Fictional B2B SaaS workflow. Not a client result.

SCENARIO

Fictional Team Atlas is preparing workspace role management. The requirement says Workspace Managers can update member roles, but it does not decide whether a Manager may grant a role equal to or above their own, which layer enforces that rule, or what evidence proves it before release.

FAILURE CHAIN

1. **Decision missing**
   Product intent names who can change roles, but not the privilege ceiling.
2. **UI becomes the assumed control**
   An elevated role is hidden, but that presentation choice is not an authorization contract.
3. **Server behavior stays unverified**
   The team has not yet produced evidence for direct API, bulk, or legacy paths.
4. **Release evidence arrives too late**
   Reviewers cannot show that every role pair follows one approved policy.

EARLIEST BREAKDOWN

Before implementation begins

The missing control is an approved privilege-boundary decision with a named owner and required proof, not another late test case.

HIGHEST-LEVERAGE NEXT INTERVENTION

Create one approved role-pair decision matrix, enforce it at the API boundary, and require executable negative-path evidence before role-management work can enter release readiness.

WHY THIS FIRST It closes the missing decision, enforcement, and evidence path together.

FACTS / ASSUMPTIONS / UNKNOWNS

FACT The fictional requirement omits the role-hierarchy decision.

ASSUME Hiding a role in the UI may be treated as enforcement.

UNKNOWN API policy, alternate update paths, and audit-log coverage.

BEST POSITIONED TO EXECUTE

- **Client Product + Security**
  Approve the hierarchy, exceptions, and release owner.
- **Client Engineering**
  Implement the API policy and role-pair tests.
- **KingBee, optional**
  Build an evidence gate only if the Blueprint shows it is warranted.

ACCEPTANCE AND EVIDENCE

- Every actor and target-role pair has an approved allow or deny decision.
- A denied direct API request returns 403 and produces no role-state change.
- UI, direct API, bulk, and legacy paths use the same policy or are explicitly excluded.
- The test result, policy owner, approval, and material denial event remain inspectable.

WHAT A REAL BLUEPRINT ADDS

Client interviews and artifacts, a baseline or measurement plan, complete risk and failure-path analysis, target-state workflow, implementation options, responsibilities, access needs, exclusions, and fixed KingBee scope only when justified.