MODERN INFRASTRUCTURE & IAC

IDEMPOTENT ISN'T AUDITABLE

Your pipeline can reproduce a run. Can it prove the run was authorized?

SWIPE TO SEE THE GAP →

What idempotency actually guarantees

✓ WHAT IT GUARANTEES

Same input → same output

Re-runs produce identical state

Drift is detectable

Rollback is reproducible

✗ WHAT IT DOESN'T PROVE

The change was authorized

Policy was current at execution

The plan matched what was reviewed

Anyone can reconstruct why

Why teams confuse them

THE PIPELINE SAYS:

- ✓ Plan succeeded
- ✓ Apply succeeded
- ✓ State file matches
- ✓ CI/CD pipeline is green
- ✓ Drift detection shows nothing

NONE OF THIS ANSWERS:

- → Who authorized this?
- → What policy was active?
- → Was the plan the same as reviewed?
- → Can you prove it 6 weeks later?

Green pipeline = successful execution. Not authorized execution. These are different.

The four auditability gaps

01 CHANGE PROVENANCE

Who triggered it, from what state, at what point in the approval chain

02 INTENT CAPTURE

The approved intent at review time — preserved as an artifact, not just a Git commit

03 POLICY STATE AT EXECUTION

What constraints were active when the apply ran — not when the plan was generated

04 EXECUTION EVIDENCE

A record linking the applied plan to the authorization event that survives beyond the run

GitOps widens the gap

Plan generated

Policy exception

Apply executes

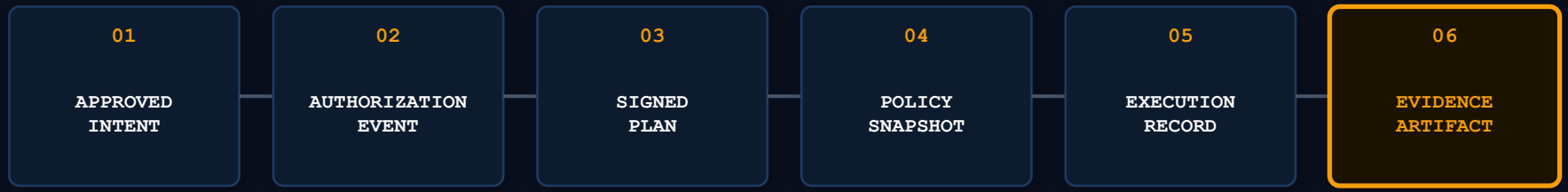
THE SCENARIO

A Terraform change opens a security group. Plan reviewed Tuesday. Apply runs Thursday.
Between those events, a governing policy exception is removed. The apply succeeds.
Six weeks later: Who approved it? What policy governed it? Was the plan consistent with what was reviewed?
The logs are gone. The state file exists. Nobody can answer questions 2 or 3.

The PR merge is treated as authorization. It captures intent at review time — not at execution time.

Framework #151 Infrastructure Evidence Gap

Auditability is not a log retention problem. It's a chain-of-custody problem.



FAILURE STATE — EXECUTION WITHOUT PROOF

The pipeline is green. The authorization chain is gone.

Doctrinal parallel: #149 AI Evidence Artifact Layer — same principle, different execution domain

DIAGNOSTIC

If your Git provider, Terraform backend, and CI/CD platform disappeared tomorrow — could you still prove who approved your last production change, what policy governed it, and what actually executed?

The last decade:

Making change deterministic.

The next decade:

Making change defensible.

Full architecture breakdown — Framework #151 Infrastructure Evidence Gap

rack2cloud.com/infrastructure-auditability/

Follow The Architect · rack2cloud.com