
USE CASE

From architecture diagram to a fix you can check

Aribot follows one path from your architecture to an approved, re-checked fix, and keeps the evidence for each step at the step that produced it.

THE PROBLEM

Our threat model lives in a document and our scanner lives in the pipeline, and nothing carries a finding from one to the other.

Leadership gets different numbers for the same estate. By the time a finding reaches a report, nobody can say which threat or resource produced it.

WHY IT MATTERS NOW

60%

of organisations managing risk in silos had a breach,
against 41% of those using integrated GRC automation

Source: Hyperproof 2025

HOW ARIBOT HANDLES IT

1 **Model the architecture**

Upload a diagram or build one, and its components, data flows and trust boundaries become the structure every later step refers to.

2 **Derive threats per component**

Each threat is attached to the component it applies to, with the reasoning kept so a reviewer can challenge it.

3 **Map threats to controls**

Controls are expressed against the frameworks you report on, so adding a framework reuses the existing mapping.

4 **Test controls in the cloud**

Read-only access to connected accounts lets each scan result attach to a specific control and resource.

WHAT YOU GET

- Findings from read-only access, with nothing in your estate changed until a person approves a specific action
- An evidence pack that names the control, the resource tested and the date
- Posture scores that list what was not measured
- One record linking each threat to its control, resource and fix

SEE IT IN THE DEMO

- A guided screen that runs from creating the workspace to exporting the evidence
- A step for connecting a cloud account with read-only access
- A findings list where untested items carry a NOT TESTED label
- An approval step where a reviewer approves or refuses each proposed fix

See it for yourself

ayurak.com/demos/how-it-works

€600 per diagram, three regenerations included · free
Starter tier with one diagram