native

NATIVE.SECURITY

Cloud Hardening — A Major 2027 Initiative

Secure-by-design for the AI era

Cloud hardening brings prevention and containment into the architecture so teams can accelerate AI adoption without leaving more access, permissions and paths available than the business actually needs.

THE CLOUD HARDENING TEST

Six questions worth asking before your 2027 plan is locked

- Can you keep an AI agent in test from reaching production unless that access is explicitly approved?
- Can you remove third-party access that sits outside the approved path?
- Can you see what a preventive control will affect before you turn it on?
- Can you stop a workload from crossing a segmentation boundary it was never meant to cross?
- Can you keep privileged actions inside the conditions you intended?
- Can you keep a sensitive account off the public internet?

If the answer is unclear, that is the gap worth looking at.

WHY TEAMS HAVEN'T HARDENED YET

01 **They're still in monitor mode.**

Policies sit in audit, controls exist but are not attached, and reporting is doing the job enforcement should be doing.

02 **The team that wants the boundary doesn't own the enforcement point.**

The control may live at the management account, organization or network layer, while the team accountable for the risk cannot safely change it.

03 **Nobody wants to guess in production.**

The environment is already running, and the cost of blocking the wrong identity, workload or pipeline can be real. Prevention stalls when impact is unknown.

04 **Exceptions quietly become policy.**

Every rollout creates edge cases. Without owners and expiration dates, temporary exceptions accumulate until the intended boundary no longer means much.

WHERE TO START

01 **Map what is actually enforced today.**

Start with the controls the cloud is enforcing now, not the policies you intended to deploy.

02 **Pick two boundaries that matter.**

Tie them to something real, such as agents moving toward production, a new cloud, a data perimeter requirement or an audit finding.

03 **Simulate against real activity.**

See which identities, workloads and pipelines a proposed control would affect before enforcement changes production.

04 **Enforce in a bounded place first.**

Start with a new organization, a non-production OU, one account group or the cloud you just adopted, then expand from there.

05 **Make exceptions temporary and visible.**

Tell teams why something was blocked, give exceptions an owner and expiry date, and restore controls when they drift.

WHERE NATIVE COMES IN

Native turns security intent into provider-native controls across AWS, Azure, Google Cloud and OCI, simulates impact against real cloud activity before rollout, and keeps the enforced architecture aligned as the environment changes. The point is not another dashboard. It is a safer way to move from intended policy to prevention and containment the cloud actually enforces.

Want to pressure-test this against your environment?