ACPG Concept

“ACPG automatically generates compliant code and AI outputs, proves they’re compliant, and attaches a cryptographically signed certificate for audit and deployment.”

Imagine **every piece of code, every AI model answer, every configuration file** your teams produce going through a smart, automated *compliance governor* before it ever reaches production. The concept is not for a reviewer or checklist. A governor or judge.

Compliance is manual, slow, error-prone, and easily overwhelmed by AI-generated output.

Our risks go up, our cost goes up, and our audit posture is reactive instead of proactive. ACPG fixes that.

What ACPG Actually Does

Think of ACPG as a **digital compliance courtroom** built into your DevSecOps pipeline.

There are three automated actors:

1. The Generator or Developer

This is the coder or the AI model that writes code, config, or documents. It tries its best to follow the rules, but, being human or (especially) AI, it gets things wrong.

2. The Prosecutor

This part behaves like an automated offensive security team and auditor. It tries to *break* the thing you just created.

It will:

• run security tests

• scan for privacy issues

• use static analysis

• generate worst-case scenarios

• look for compliance breaches

If there is a way to violate a policy, it will find it.

3. The Judge (Adjudicator)

This bit is the brain.

It weighs the evidence from both sides and determines whether the artefact follows **all** the rules.

It uses formal logic so it can:

• handle exceptions

• resolve conflicting policies

• understand priorities (law > internal guideline > cosmetic rule)

• explain *why* something is compliant or not

What Happens Next

If the artefact is *not* compliant, the judge tells the generator **exactly what to fix**.

Then the generator automatically produces a new version.

The prosecutor attacks again.

The judge reviews again.

This loop continues until the artefact is **fully compliant**.

The End Product: A Proof-Carrying Artefact

Once the artefact passes every rule:

✓ A **machine-readable compliance certificate** is attached

✓ It includes the rules that applied

✓ The evidence

✓ The attacks raised

✓ How they were resolved

✓ A cryptographic signature tying it all to that exact version

The result is something with embedded cryptographically signed evidence for auditors and regulators:

An artefact that proves its own compliance without anyone having to re-run tests.

Why This Matters

1. Audit Ready, All the Time

Every release, every model, every line of code comes with its own compliance proof. No more investigation and testing before audits.

2. Risk Reduction

Instead of catching problems at the end, ACPG fixes them **as the code is being written**. We lower the chance of breaches, fines, and brand damage dramatically.

3. Works at AI Scale

AI will generate more code and content than any human team can review. ACPG automatically reviews *every* generation. Human reviewers can't keep up.

4. Future Regulatory Shield

Regulators (especially in AI) are moving toward:

• traceability

• explainability

• provable compliance

ACPG gives us all three.

5. Cultural Shift: Compliance by Construction

Instead of slowing development down, compliance becomes part of the build.

We ship faster, safer, and with far less manual effort.