Open Policy Agent Decoupled Policy From Code. The Policy Is Not Cryptographically Bound.

by Nick Clark | Published March 27, 2026 | PDF

Open Policy Agent established policy-as-code as a standard practice by decoupling authorization decisions from application logic. Write policies in Rego, evaluate them against structured input, and receive allow/deny decisions. The decoupling is valuable. But OPA evaluates policy at decision points without cryptographic binding. Policy decisions are not signed, not bound to the operations they authorize, and not persisted as cryptographic governance lineage. The gap is between policy evaluation and cryptographic governance.

OPA's impact on the policy landscape is significant. Making policy a first-class artifact that can be versioned, tested, and deployed independently from application code was a genuine advance. The gap described here is about the binding between policy decisions and the operations they govern.

Policy evaluation is advisory until enforcement makes it binding

OPA evaluates a query against a policy and returns a decision. The calling application receives the decision and acts on it. But the decision is a data structure returned by an API call. It is not a cryptographic artifact.

A compromised application can ignore an OPA deny decision. A misconfigured enforcement point can skip the OPA check entirely. The policy evaluation is structurally advisory. Enforcement depends on every integration point correctly implementing the check and honoring the result.

Decision logs record, not bind

OPA's decision logging records every policy evaluation: the input, the policy version, and the decision. This is valuable for auditing. But the decision log is a record of what OPA was asked and what it answered. It is not a cryptographic binding between the decision and the operation that followed.

If the application executed an operation after receiving a deny decision, the decision log shows the deny. But the log does not prevent the operation. Governance and enforcement are separate systems that must be manually kept in sync.

What cryptographic governance provides

Cryptographic governance makes policy binding structural. Every operation carries a signed policy reference. Every mutation is gated by cryptographic validation that the policy authorizes the specific operation in the specific context. The binding is not advisory. It is cryptographic. An operation without a valid signed policy cannot execute because the execution layer requires it.

Decision provenance is recorded in lineage as part of the operation's cryptographic history. The governance decision is not a separate log entry. It is an intrinsic part of the operation's audit trail, signed and verifiable.

The remaining gap

OPA made policy a first-class artifact. The remaining gap is in binding: whether policy decisions are cryptographically bound to the operations they govern, making governance structural rather than advisory.

Cryptographic Governance All 21 steps →

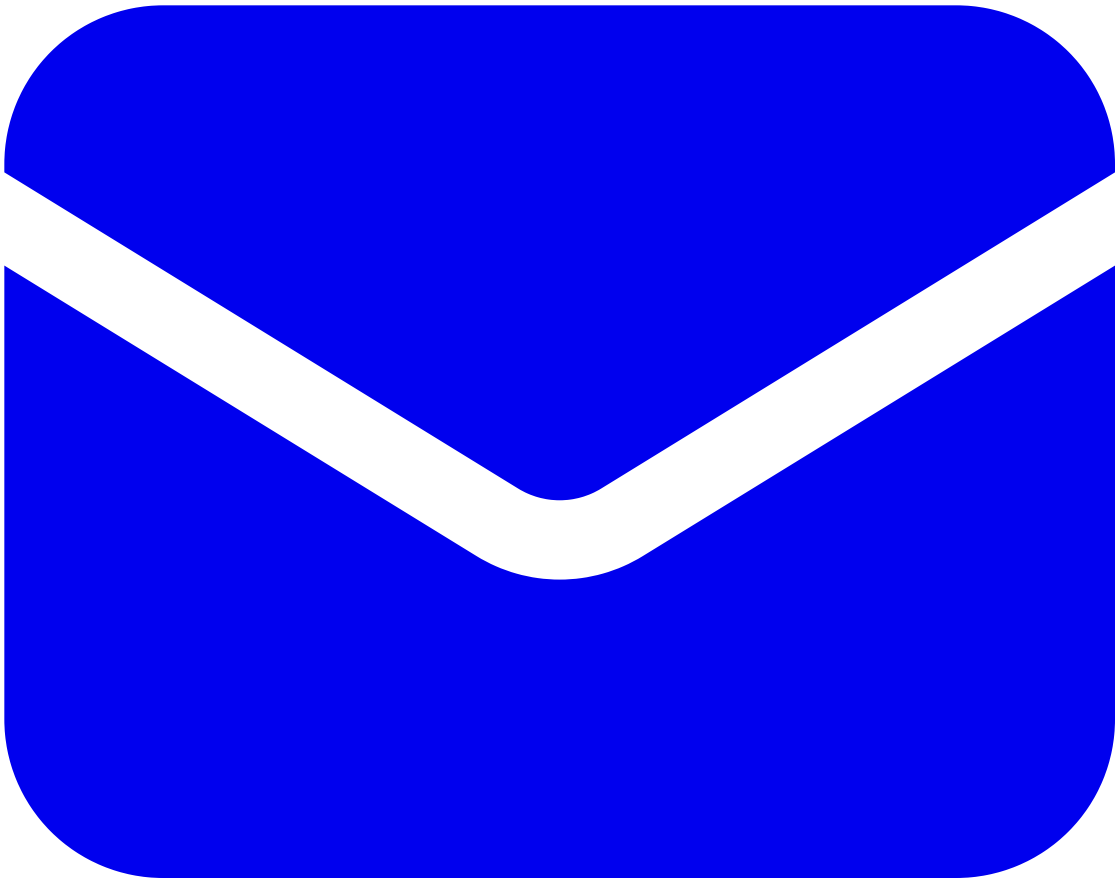
Policy that binds cryptographically — not by convention.

Patent
US 19/561,229 · filed
Primary Technical Disclosure
○ Ethical Enforcement as Infrastructure: Cryptographic Governance for Autonomous Systems
Secondary Technical
○ Governance Gate as Deterministic Precondition: No Verification, No Execution○ Canonical Alias to External Policy Indirection: Policy Evolution Without Agent Mutation○ Immutable-by-Default Policy Objects: Governance Changes Through Successor Issuance○ Runtime Policy Resolution Pipeline: Mandatory Verification Before Every Execution○ Freshness, Revocation, and Anti-Rollback Controls: Preventing Stale Authority○ Memory-Derived Eligibility Conditioning: Past Violations Constrain Future Authorization○ Intent-Independent Authorization: Governance Without Alignment Scoring○ Execution Feedback as Enforcement Signals: Operational Outcomes Shaping Future Authorization○ Trust Degradation as State Transition: Policy-Defined Narrowing of Permitted Actions○ Structural Quarantine: Execution Prevention Until Authorized Remediation○ Lineage-Constrained Governance Inheritance: Constraints That Persist Across Generations○ Unauthorized Fork Prevention: Lineage Continuity as Anti-Cloning Mechanism○ Meta-Policy Objects: Higher-Order Constraints Across System Behavior Categories○ Quorum-Based Governance Override: Multi-Party Approval With Signature-Chain Continuity○ Distributed Alias Publication: Policy Dissemination Through Federated Registries○ Fallback Enforcement Agents: Distributed Monitors as Defense-in-Depth○ Append-Only Governance Audit Ledger: Tamper-Evident Records of Every Authorization○ Governance Without Persistent Keypairs: Trust-Slope Authorization Replacing Static Keys○ Execution Eligibility Indicator: Dynamic Computation From Policy, Memory, and Lineage
Applications (General)
○ EU AI Act Compliance Through Structural Governance○ Financial Services Audit Trails Without Trusted Intermediaries○ Healthcare Compliance Through Structural Governance○ Defense Data Classification Enforcement○ Environmental Monitoring With Tamper-Proof Governance○ Pharmaceutical Supply Chain Governance○ Nuclear Facility Operational Governance○ Child Safety Content Enforcement
Applications (Specific)
○ HashiCorp Vault Manages Secrets. It Does Not Make Policy Cryptographically Binding.○ AWS KMS Manages Encryption Keys. The Keys Do Not Carry Governance.● Open Policy Agent Decoupled Policy From Code. The Policy Is Not Cryptographically Bound.○ Styra Made OPA Enterprise-Ready. The Governance Model Did Not Change.○ Snyk Finds Vulnerabilities Before Deployment. Governance After Deployment Is Still Manual.○ Palo Alto Networks Inspects Traffic. It Does Not Govern the Operations That Generate It.○ SPIFFE/SPIRE Provides Workload Identity. The Identity Has No Cryptographic Governance Binding.○ cert-manager Automates Certificate Lifecycle. The Certificates Carry No Governance Policy.○ Keycloak Provides Open-Source Identity Management. The Tokens It Issues Carry No Governance Binding.○ HashiCorp Boundary Provides Zero-Trust Access. The Access Sessions Have No Cryptographic Governance.○ Teleport Provides Unified Infrastructure Access. Access Control Is Not Cryptographic Governance.○ BeyondTrust Manages Privileged Access. Privilege Is Not Cryptographic Governance.○ CyberArk Pioneered Privileged Access Security. The Privilege Model Has No Cryptographic Governance Layer.○ 1Password Made Password Management Accessible. The Credentials It Manages Are Still Credentials.
Cryptographic Governance overview →
AQ
deterministic
autonomy

Legal

Subject to one or more pending U.S. and international patent applications, see Patents for the current list and status. No license, express or implied, is granted. Any use requires a separate written agreement—see Licensing. Patent applications referenced on this site are pending. Claim scope, if any, is subject to examination and may issue in altered form or not at all. See Legal for terms and conditions.

Last updated: 2026-03-03



-
- nick@qu3ry.net
- 72 28 14 36 01

Invented by Nick Clark | Founding Investors: Devin Wilkie