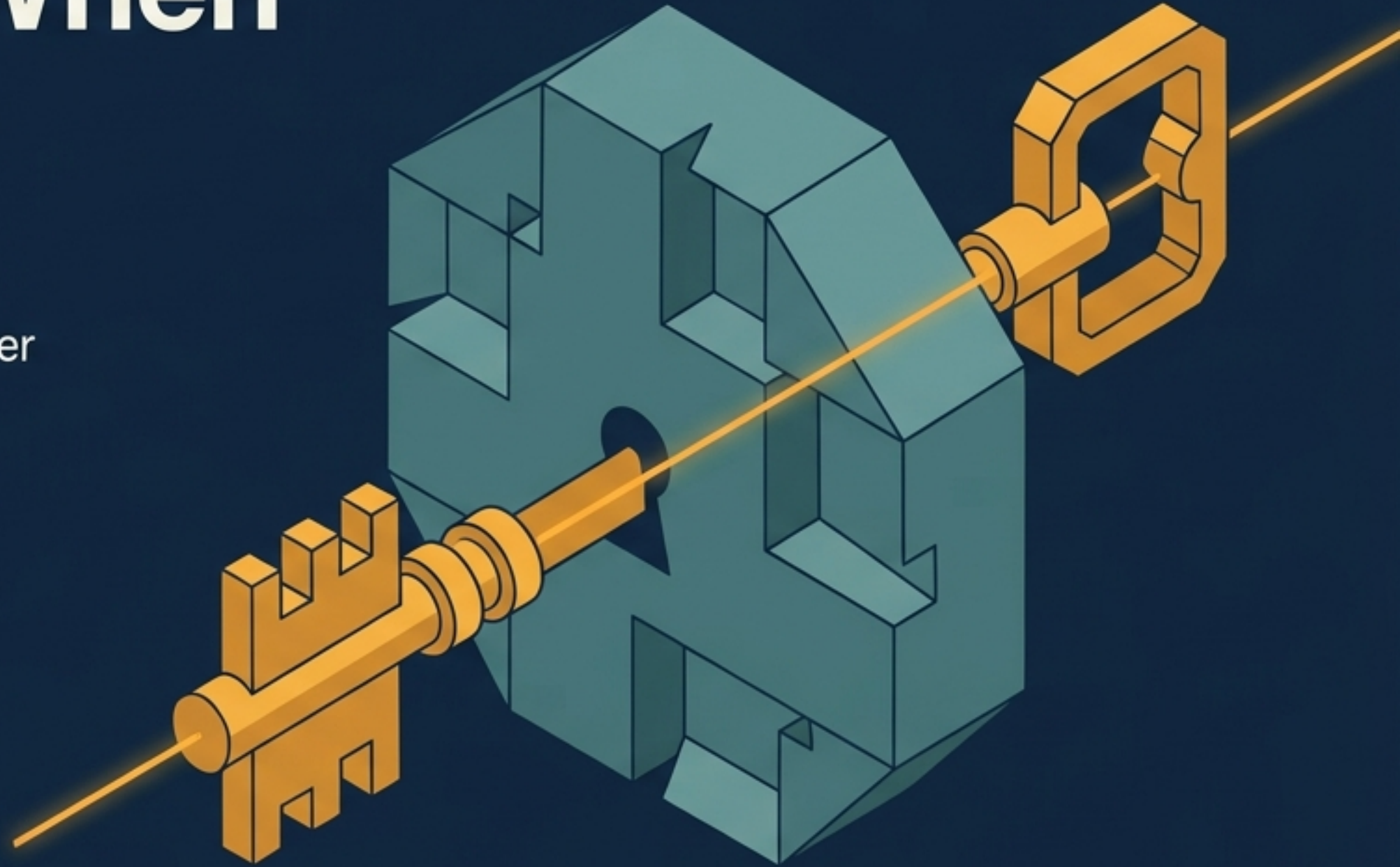


Your AI Agent Has an ID. Can It Keep One When Its Keys Change?

An AI agent in a software pipeline is more than a clever assistant; it is a delegated authority interacting directly with your repositories, data, and tools.

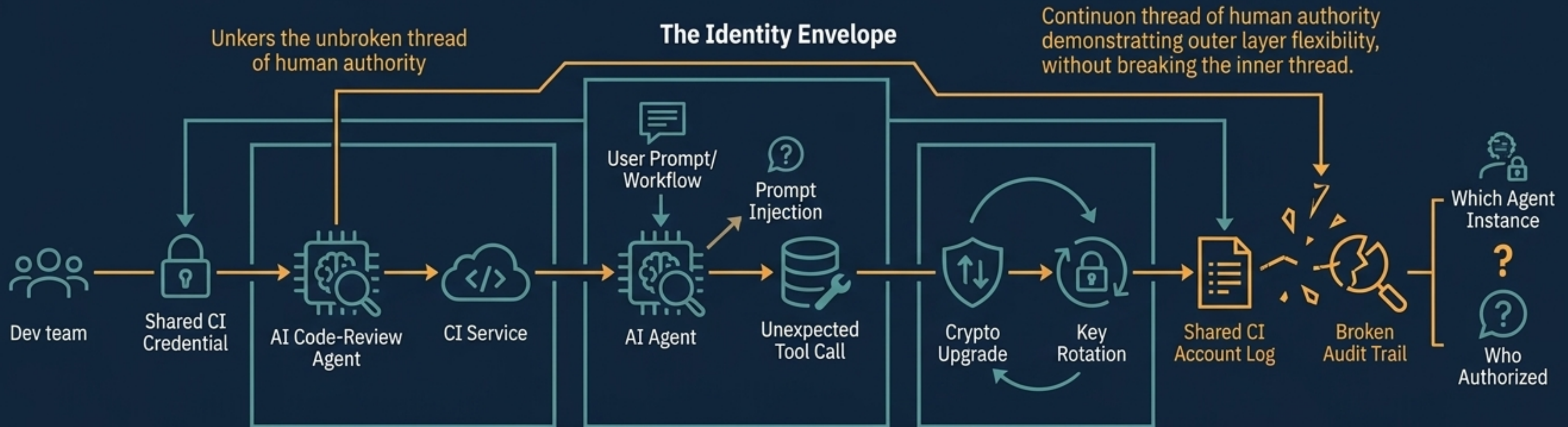
The security question for Singapore enterprise leaders is no longer just “Is the model safe?” It is: **Which non-human identity did the work, who delegated authority to it, and can that identity be audited and replaced when cryptographic algorithms evolve?**



The Hidden Risk in Delegated Agent Architectures

HYPOTHETICAL DESIGN
SCENARIO, NOT A
REPORTED INCIDENT.

When agents share credentials, routine cryptographic maintenance obscures accountability. If an agent's permission model, human delegation record, and signing-key implementation are tightly coupled, a simple key rotation can break your audit trail.



Step 1: Dev team gives an AI code-review agent a shared CI service credential.

Step 2: A prompt injection or misconfigured workflow triggers an unexpected tool call.

Step 3: The team rotates a signing key during a wider crypto upgrade.

The Break: The logs only show the shared CI account. The team has rotated the key but cannot answer which agent instance acted or who authorized it.

The Evolving Standard for Agent Accountability

Agent security is actively moving from general principles toward concrete implementation examples.

NIST (U.S.)

29 Sep 2026: NCCoE announces its first DevSecOps implementation use case to demonstrate agent identification and authorization.

Caveat: Driven by the February 2026 NIST concept paper. This is a draft implementation project exploring open engineering questions—not a final, binding standard.

Singapore

17 Jun 2026: Cyber Security Agency (CSA) publishes agent-specific guidance requiring mapping of agentic workflows (e.g., coding assistants, fraud detection) to implement lifecycle controls.

20 May 2026: IMDA Gov-Google Sandbox on computer-use agents highlights that traditional identity, authentication and access-control foundations may need to evolve for autonomous interactions.

Caveat: This sandbox is exploratory; it is not evidence of ubiquitous production deployment.

Disentangling the Layers of Agent Access

An opaque bearer token, a signed token, and a certificate-based workload identity do not have identical properties. To maintain control, these four layers must work together, but one layer must never stand in for the others.

Identity (Who)

Which specific workload or agent instance acted?

Authorisation (What)

What is it permitted to do, and for whom is it acting?

Cryptography (How)

The mechanism proving and protecting the exchange (e.g., TLS, signing key).

Auditability (Record)

Preserving the verifiable chain connecting the action back to the human sponsor.

Quantum Preparation is Planned Migration, Not Immediate Panic

An algorithm transition must not silently change who an agent represents. The goal is Crypto Agility: adapting or replacing cryptographic algorithms across infrastructure while preserving security and ongoing operations.

The Myth

- Installing new algorithms everywhere today.
- Treating "PQC-ready" as a substitute for strong agent access controls.
- Assuming immediate blanket key replacement is mandated for all Singapore businesses.
(Note: PQC is not yet mandatory for all SG businesses).

The Reality

- Knowing which cryptographic component does what.
- Separating cryptography from business identity where feasible.
- Safe testing of dependency changes in non-production environments.

Singapore's QRI and Handbook define this as a complex, multi-year effort. NIST's first three PQC standards are ready, but implementation depends on your actual cryptographic dependencies and vendor support.

The SCOPE Framework for Accountable Agents

Use this five-part design check before an AI agent receives access to code, cloud resources, customer records, or operational workflows.



Sponsor every agent

Assign a named business and technical owner. A model name is not an accountable owner.

Constrain each task

Use distinct workload identities. No shared admin accounts. Do not let prompt text silently expand permissions.

Observe authority chain

Log the context (initiating user, agent ID, policy decision, target resource) without placing secrets in the logs.

Pause and revoke

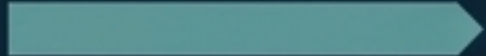
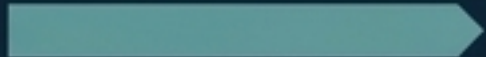
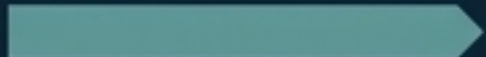
Implement and test revocation paths. An untested kill switch is only a design assumption.

Evolve cryptography

Map cryptographic dependencies. Record the algorithm, owner, and replacement path—not the secret key material.

The 30-Day Operational Remediation Roadmap

Track outcomes that prove control rather than activity. The goal is zero unowned or shared credentials for high-impact agents.

Timeframe	Action	Accountable Owner
First 72 Hours 	Identify high-impact agents (code, cloud, data). Record sponsor, principal, and credentials. Add human checkpoints if authority is unclear.	CIO/CTO Sponsor
Days 4–10 	Replace shared or ownerless credentials with distinct workload identities. Remove broad grants.	IAM / Cloud-Platform Owner
Days 11–20 	Connect logs across the user, agent, and target. Define approval gates. Exercise pause and credential revocation.	SOC / Engineering Leads
Days 21–30 	Map crypto components. In a test environment, validate algorithm rotation and interoperability aligned with CSA QRI and NIST.	PKI / Enterprise-Architecture Lead

Source Note: KIS AI Security briefing.

Make the Identity Accountable Before Making the Agent Autonomous.

Agentic AI and post-quantum migration are operationally dependent. Both rely on knowing which digital identity is trusted, what it may do, which cryptographic mechanism supports it, and how change is controlled.

If you are introducing agents into development or cloud workflows, review the authority, identity, logging, and change paths around those systems. Engage KIS AI Security Services to conduct source-code assurance, prompt-injection testing, and architectural identity reviews.