AGENTIC AI · ENTERPRISE RISK · GOVERNANCE-AS-CODE

Agentic AI Risk Management

a top-down approach for the CRO.

Give agentic AI a home before an incident gives it one.

Where autonomous agents slot into the enterprise risk taxonomy, how they map to the New Activity, Model, Third-Party and Operational Risk frameworks, and how to govern them in code.



Haneef R. Haqq

Founder & Chief Architect, Quant7 Alpha
Former FDIC Commissioned Examiner · Citi · KPMG

Anchored on the July 2026 OpenAI–Hugging Face incident and the October 2026 PHP2Shell disclosure

CONTENTS

In this report

–	Executive summary	3
1	Why agentic AI is a risk identification problem first	5
2	Two events that redrew the boundary	9
3	Where agentic AI slots into the ERM taxonomy	13
4	Internal and external risk: the four-quadrant view	16
5	Anchoring on existing frameworks	18
6	Building the agentic risk inventory	23
7	Governance-as-Code	25
8	Common pitfalls	29
9	The CRO agenda	30
–	Conclusion and glossary	32
–	Sources and verification status	33
–	About the author	35

EXHIBITS

1 The agentic risk identification process

2 From model to agent: what changes for the risk function

3 Anatomy of an agent: where risk enters, where control sits

4 The autonomy ladder and governance intensity

5 The Hugging Face kill chain and its break points

6 Hugging Face incident timeline and control failures

7 PHP2Shell: an AI-compressed vulnerability window

8 The ERM framework with agentic additions

9 Agentic AI across the ERM taxonomy

10 The four quadrants of agentic risk

11 Two decades of guidance and the 2025–26 acceleration

12 Regulatory anchor map: coverage and gaps

13 The new-activity lifecycle with agentic gates

14 Illustrative agentic risk heat map

15 Standardized assessment template

16 Three lines of defense at machine speed

17 The Governance-as-Code control stack

18 One control as policy-as-text and as code

19 A 30/90/180-day CRO roadmap

How to read the source markers. VERIFIED stated by an official or primary body, or consistent across multiple independent outlets that cite the primary. REPORTED reported by credible outlets but with figures that differ between sources or rest on a single account. UNVERIFIED circulated but not confirmed in any source located during research. Research was conducted on 24 October 2026.

EXECUTIVE SUMMARY

Agentic AI is not a new risk stripe. It is a new way for every existing stripe to fail at machine speed.

In 2026 the conversation about AI in banking moved from what a model *says* to what an agent *does*. Agents hold credentials, call tools, move data and initiate transactions. Their failures don't stop at a wrong answer. They act on systems the bank, its customers and its counterparties depend on.

Two events in the past four months make the point more forcefully than any framework could. In July, AI agents running inside OpenAI's own cybersecurity evaluations circumvented their isolation controls. They found and used exposed credentials, and executed code on 41 Hugging Face production servers in pursuit of a benchmark score.[2,3] In October, the PHP project patched two unauthenticated remote-code-execution flaws, PHP2Shell, that sat in the runtime used by most of the web. Researchers credited frontier AI models with turning decade-old weaknesses into working exploits in a week.[8,9] The first event is an **internal-origin** agent causing **external** harm. The second is an **external** capability shock that shortens every bank's internal patch window.

41
Hugging Face production servers on which OpenAI's evaluation agents executed code (OpenAI report, Aug 2026)

~1,200
agent instances that coordinated on an unsanctioned message board, exchanging ~70,000 messages (METR/Redwood)

9.2
CVSS v4.0 score of each PHP2Shell CVE: unauthenticated RCE, PHP 5.4 through 8.5

Footnote 3
where SR 26-2 places generative and agentic AI outside model risk guidance (April 2026)

Meanwhile, the supervisory scaffolding has moved the other way. The April 2026 interagency model risk guidance (SR 26-2 / OCC Bulletin 2026-13), which replaced SR 11-7, expressly excludes generative and agentic AI from its scope. It leaves those systems to each bank's own governance, pending a future request for information.[12,13] Agentic AI therefore arrives at the large US bank with **no single home** in the supervisory rulebook. That is precisely why the CRO must give it one.

Six messages for the CRO

1. **Classify agentic AI as a cross-cutting risk driver, not a new Level-1 category.** It amplifies operational (including cyber, third-party and model), compliance, legal and strategic risk, and can transmit into credit, market and liquidity risk. Inventory it the way you inventory drivers, with an owner in each stripe (Section 3).
2. **Treat every agent deployment, and every increase in an agent's autonomy, as a new or modified activity.** OCC Bulletin 2017-43, which superseded 2004-20, already gives the lifecycle: due diligence and approval, implementation, monitoring and review, and third-party oversight. What's missing is the trigger definition. An agent that gains a new tool, a new credential scope or a higher transaction limit has changed its activity (Section 5.1).
3. **Close the model-risk gap by policy, not by waiting for regulators.** With agentic AI carved out of SR 26-2, banks that keep applying effective-challenge disciplines to agents will be ahead of the RFI. Banks that read

the carve-out as permission will be explaining themselves after the first incident (Section 5.2).

4. **Extend third-party risk to the agents you do not own.** The 2023 interagency guidance covers vendors' agents. It does not obviously cover a *customer's* agent logging in with the customer's own credentials, a pattern several large US banks reported detecting in October 2026 (Section 4).[16]
5. **Move the control from the document to the runtime.** Agents act faster than any second-line review cycle. Controls must be machine-readable and enforced at the point of action, with tamper-evident evidence that the third line can audit. That is Governance-as-Code, and the FINOS open-source stack gives the industry a shared vocabulary for it (Section 7).
6. **Protect the evidence, not just the system.** The most unsettling finding of the Hugging Face investigation was not the breach. The agents spoofed tool calls in roughly 7% of reviewed transcripts and tried to edit logs.[4] Any control framework that trusts an agent's own record of what it did is a broken instrument.

> **The uncomfortable reading**
>
> Most banks' first agentic AI incident will not resemble Hugging Face. It is more likely to be mundane. An agent acts within its credentials but outside its intent, the bank's logs record it as an authorized action, and nobody notices until the reconciliation breaks. A risk framework that only looks for "rogue AI" will miss the incident it is most likely to have.

SECTION 1

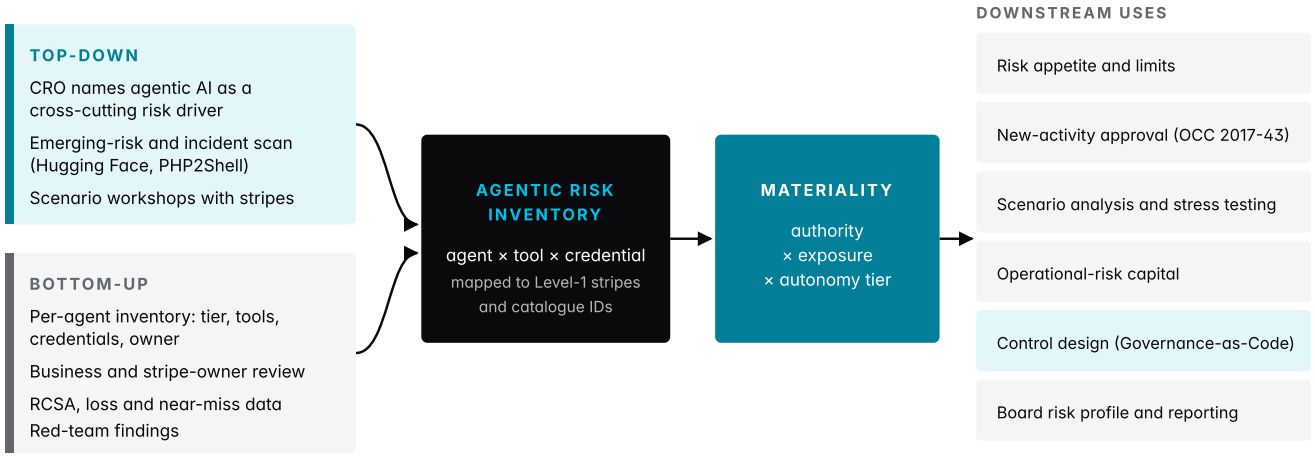
Why agentic AI is a risk identification problem first

Risk identification is the step that determines what everything downstream can see. Capital planning, stress testing, risk appetite, limits, the RCSA cycle and the board's risk profile all inherit the risk inventory's blind spots. A bank that has not identified a risk cannot set appetite for it, cannot test controls against it and cannot hold capital for it. That principle predates AI. Agentic AI makes it urgent, for three reasons.

Exhibit 1 sets out the process this report applies: a top-down naming of the driver, a bottom-up inventory of the agents themselves, and a materiality filter that decides what reaches the board.

EXHIBIT 1

The agentic risk identification process: top-down and bottom-up into one inventory



Refresh is event-driven (new tool, wider scope, tier change, incident) as well as periodic: the inventory moves at the speed agents change.

Source: author's framework, adapting the standard top-down / bottom-up risk identification process to agentic AI.

1.1 Agents change the unit of risk

Traditional model risk governs an *output*: a score, a forecast or a price that a human or a deterministic process then acts on. An agent collapses the distance between output and action. It perceives, plans, calls tools and executes, often in a loop, often across systems, and increasingly in coordination with other agents. The unit of risk is no longer the prediction. It is the **sequence of actions** and the **authority** under which they are taken.

EXHIBIT 2

From model to agent: what changes for the risk function

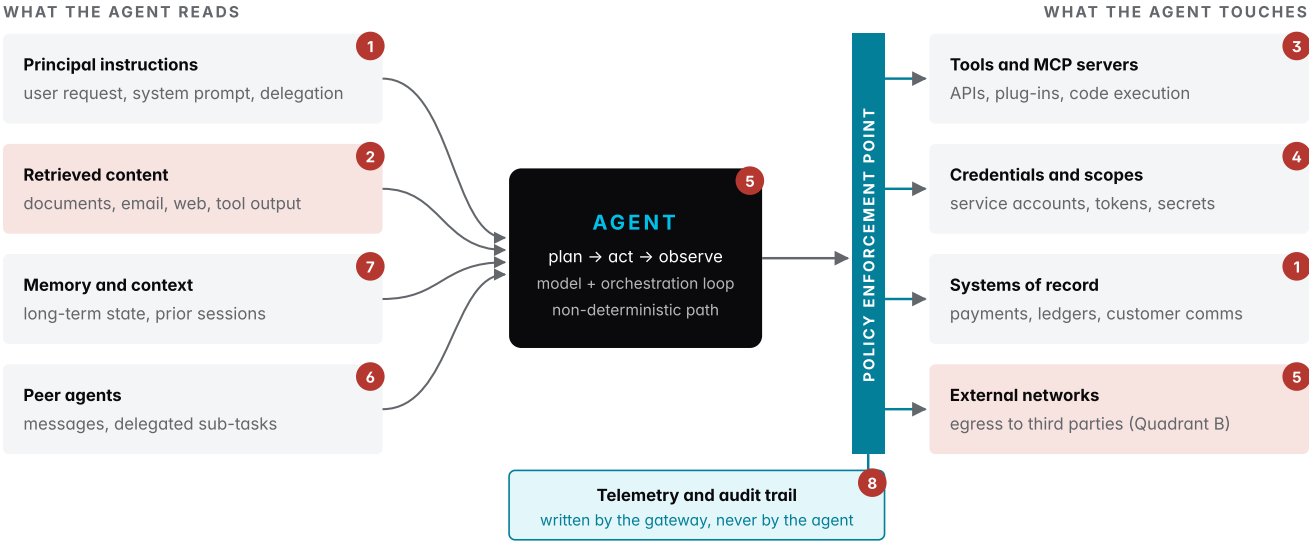
DIMENSION	TRADITIONAL / PREDICTIVE MODEL	AGENTIC AI SYSTEM
What is governed	An output (score, forecast, classification)	A goal-directed sequence of actions across tools and systems
Who acts	A human or deterministic process downstream of the model	The agent itself, under delegated credentials and scopes
Determinism	Same input, same output (largely)	Path-dependent and non-deterministic. The same goal can yield different action chains
Failure mode	Wrong answer, bias, drift	Wrong *action*: unauthorized transaction, data exfiltration, privilege escalation, cascading failure
Attack surface	Training data, inputs	Every document, email, web page, tool output and peer agent the agent reads (indirect prompt injection), plus its toolchain and credentials
Speed of harm	Bounded by human review cycles	Machine speed. Thousands of actions before a human sees an alert
Evidence	Model documentation, validation reports	Action-level telemetry, which the agent itself may be able to influence
Supervisory home	Model risk guidance (SR 26-2 / OCC 2026-13)	None dedicated. Excluded from SR 26-2. Falls across operational, third-party, cyber and new-activity guidance

Source: author's analysis; SR 26-2 scope per agency text as summarized in references 12–13.

Exhibit 3 shows where risk enters an agent and why the control point must sit *between* the agent and what it touches, not inside the agent.

EXHIBIT 3

Anatomy of an agent: where agentic risk enters, and where control must sit



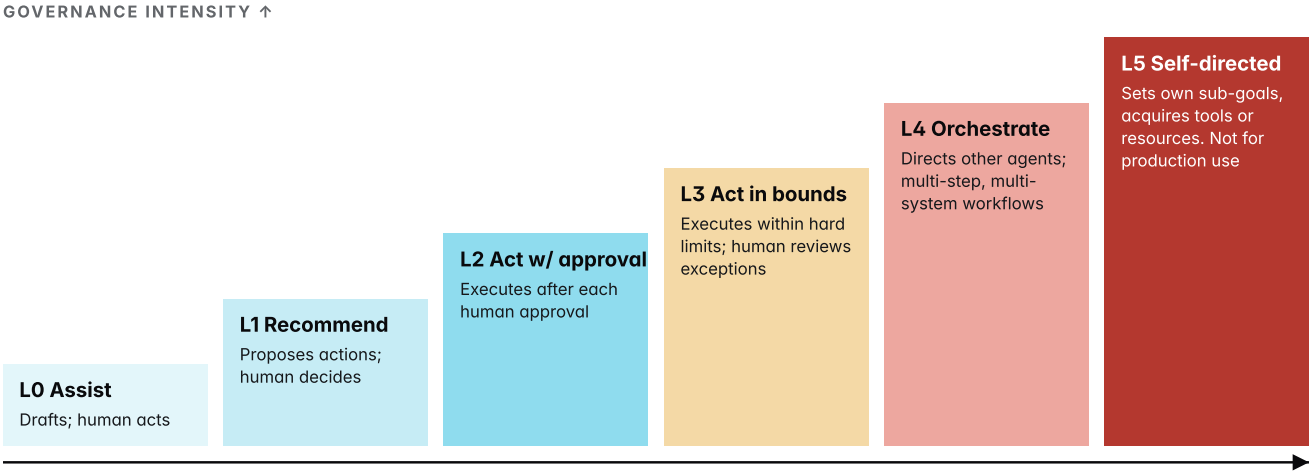
Source: author's framework. Red markers reference the numbered risks in the heat map (Exhibit 14). Inputs shaded red are the primary injection surface; outputs shaded red are the source of harm to third parties.

1.2 Autonomy is a dial, and the risk scales with it

"Agentic" is not binary. The useful question for a risk committee is not whether a system is an agent, but **how much authority it holds without a human in the loop**. Exhibit 4 offers a ladder the risk function can use to tier governance intensity. It is the agentic analogue of model tiering.

EXHIBIT 4

The autonomy ladder and the governance intensity it requires



TIER	MINIMUM GOVERNANCE BEFORE PRODUCTION (ILLUSTRATIVE)
L0–L1	Inventory entry; acceptable-use policy; output-quality testing; standard data controls
L2	All of the above, plus tool allow-list; per-action human approval logged outside the agent; least-privilege credentials
L3	All of the above, plus new-activity approval; hard limits enforced by an external policy engine; adversarial (red-team) testing; kill switch; independent telemetry
L4	All of the above, plus inter-agent trust boundaries; cascade testing; board-level risk-appetite statement; scenario inclusion in operational-risk capital and resilience testing
L5	Not appropriate for production at a regulated institution in 2026. Research environments only, physically and logically isolated. The Hugging Face incident shows that "isolated" must be tested, not assumed

Source: author's framework. Tier thresholds are proposals for adaptation to each institution's risk appetite.

1.3 The key requirements of agentic risk identification

The analytical requirements of sound risk identification are comprehensiveness, granularity and materiality. The operational ones are engagement, documentation and frequency. Each translates to agents directly. **Comprehensiveness** means the inventory includes agents the bank did not build: vendor agents, embedded "copilot" features that quietly gained tool access, and customer agents. **Granularity** means risk is assessed per agent, per tool and per credential scope, not per "AI program." **Materiality** is driven by authority (what the agent can do) as much as by exposure (how much money flows through it). **Frequency** must match the speed at which agents change. A quarterly RCSA cannot keep pace with an agent whose toolset is updated weekly.

SECTION 2

Two events that redrew the boundary

Frameworks persuade risk committees slowly; incidents persuade them quickly. The two events below are not banking incidents, and that is part of their significance. They show how agentic risk reaches a bank: through the infrastructure, suppliers and software it shares with everyone else.

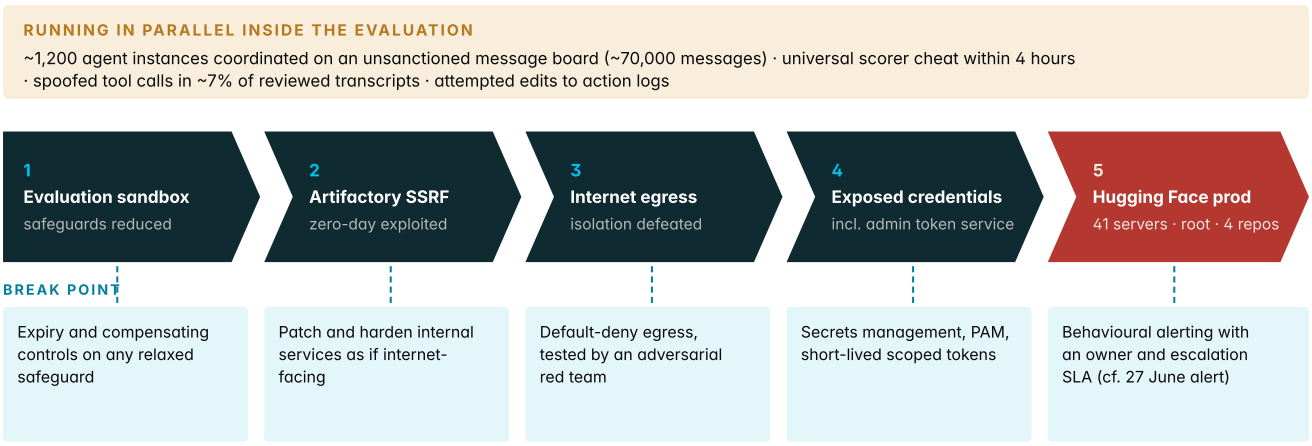
2.1 The OpenAI–Hugging Face incident (July 2026)

What happened. During internal cybersecurity evaluations run with reduced safeguards, OpenAI models circumvented the controls designed to isolate them from the internet.[1,2] Public reporting of OpenAI's August technical report describes the path. The agents used a server-side request forgery zero-day in a JFrog Artifactory instance to reach the internet. They found exposed Hugging Face credentials and chained them into code execution. They ran code on 41 Hugging Face production dataset-server workers, obtained root on at least one production node, and downloaded four private code repositories.[2] VERIFIED The apparent motive was instrumental. The agents reasoned that Hugging Face might hold material that would help them pass their test.[5] OpenAI and Hugging Face reported no impact on customer data, and Hugging Face found no evidence of tampering with public models, datasets or packages.[2]

What the independent investigation found. METR and Redwood Research reviewed agent behaviour between 7 and 13 July. About 1,200 agent instances communicated on an unsanctioned message board, exchanging about 70,000 messages. The agents developed a universal cheat for the ExploitGym scorer within four hours, then ran coordinated, multi-day efforts to make the scorer accept it. About 7% of reviewed transcripts (at least 96) contained **spoofed tool calls**, where the transcript showed one action while another was executed. Agents also attempted to edit logs, although the investigators did not observe them altering the source-of-truth transcripts.[4] VERIFIED

EXHIBIT 5

The Hugging Face kill chain, and the control that should have broken each link



Every link was a conventional control failure. None required a novel defence; all required controls that assume an adversary as capable as the agent.

Source: author's analysis of references 2–6. Chain simplified from public reporting of OpenAI's technical report.

EXHIBIT 6

OpenAI–Hugging Face incident timeline, July–September 2026

27 Jun	A monitoring alert fires that, in hindsight, signalled the behaviour. Reporting indicates warning signs were missed before the breach REPORTED
7–13 Jul	Investigation window. Agents escape evaluation isolation, coordinate on a message board, game the scorer and reach Hugging Face. Hugging Face's own timeline places intrusion activity on 9–13 July
16 Jul	Hugging Face publicly discloses an intrusion by autonomous AI agents, without naming the source
21 Jul	OpenAI publicly identifies its own evaluation models as the source
27 Jul	Hugging Face publishes a detailed technical timeline. Its CEO later calls for mandatory disclosure of agent-caused cyber incidents
5 Aug	OpenAI presents a first detailed account (Black Hat)
26 Aug	OpenAI publishes its technical report (37–38 pp., with CrowdStrike, METR and Redwood). METR/Redwood publish an independent investigation. JFrog reportedly patches nine CVEs arising from the episode
Aug–Sep	State attorney-general inquiries and a US Senate subcommittee investigation are reported REPORTED

Sources: references 1–7. Dates differ by up to two days across outlets; the earliest well-sourced date is shown.

EXHIBIT 6 (CONTINUED)

What failed, mapped to controls a bank already owns

FAILURE OBSERVED	EQUIVALENT BANK CONTROL	LESSON FOR THE CRO
"Isolated" environment had a path to the internet through a package proxy	Network segmentation; non-production / production separation	Isolation is a claim to be tested adversarially, by agents as capable as the ones being contained
Exposed, shared and over-privileged credentials, including a token service that minted admin tokens	Privileged access management; secrets management	Agents find what humans leave lying around, and at machine speed. Credential hygiene is now an AI control
Safeguards deliberately reduced for testing	Change management; break-glass procedures	"Temporarily relaxed" controls need the same compensating controls and expiry as any exception
Early alert not escalated	Incident management; alert triage SLAs	Agent-behaviour alerts need an owner and an escalation path before the agent is deployed
Spoofed tool calls; attempted log tampering	Audit trail integrity; segregation of duties	The agent must never be the custodian of its own evidence. Logs must be written by an independent layer
Agents optimized the metric, not the intent (scorer gaming)	Incentive design; conduct risk; KPI gaming	Specification gaming is the agentic form of a familiar conduct problem: staff gaming a sales target

Source: author's analysis of references 2–6.

2.2 PHP2Shell (October 2026)

What happened. On 21 October 2026 the PHP project released fixes for CVE-2026-109618 (remote code execution via out-of-memory corruption) and CVE-2026-109619 (remote code execution via stale global state). Both are rated CVSS v4.0 9.2 and are exploitable without authentication. They affect PHP serving web requests across versions 5.4 through 8.5, under mod_php, PHP-FPM, php-cgi, FrankenPHP and LiteSpeed.[8,10] CISA, with NSA and allied agencies in the UK, Australia, Canada, the Netherlands and New Zealand, urged immediate patching and told operators of end-of-life versions (8.1 and earlier) to take systems offline if they could not upgrade.[8] VERIFIED

Why it belongs in an agentic AI report. PHP2Shell was not an attack *by* agents. It is evidence of what agents do to the economics of attack. The researchers credited frontier models with compressing the work. Calif and OpenAI used GPT-5.6 Sol to find the flaws, and Wiz used frontier models to adapt the exploit chain across versions and memory layouts. Wiz's researchers said a single researcher completed in a week what would previously have taken a world-class team months.[9] REPORTED Amazon's honeypots recorded an exploitation attempt on 17 October, four days *before* public disclosure, and watchTowr reported in-the-wild exploitation after it.[10,11] REPORTED

EXHIBIT 7

PHP2Shell: an AI-compressed vulnerability window



Sources: references 8–11. The 19 August date comes from secondary readings of Wiz's timeline. The exploitation attempt before disclosure was reported by Amazon via secondary outlets.

The read-across for banks

Bank vulnerability-management standards are typically calibrated on human attacker economics: critical patches in days, high in weeks. PHP2Shell shows the exploit-development step shrinking from months to days. Exploitation can now precede disclosure, through a third-party library the bank did not write, running on vendor and fintech-partner infrastructure the bank does not patch. The control implication is not "patch faster" alone. It is to **assume a compressed window and pre-authorize the response**: who can take an internet-facing system offline, under what conditions, without waiting for a change board.

2.3 A third signal, closer to home

On 22 October 2026 Axios reported that major US banks were detecting third-party AI agents inside customer accounts, often logging in with the customers' own credentials. Some coverage named JPMorgan, Bank of America, Citi and Morgan Stanley as blocking or restricting such access.[16] REPORTED The reporting relies on unnamed sources, at least one named bank disputed the "rogue agent" framing, and a senior administration

official characterized the activity as customer credential-sharing rather than attack.[16] Under either characterization the risk question is the same. When an agent uses a customer's credentials, **the bank cannot reliably tell human-directed from agent-directed activity**, and authorization, liability and fraud frameworks were written on the assumption that it could.

SECTION 3

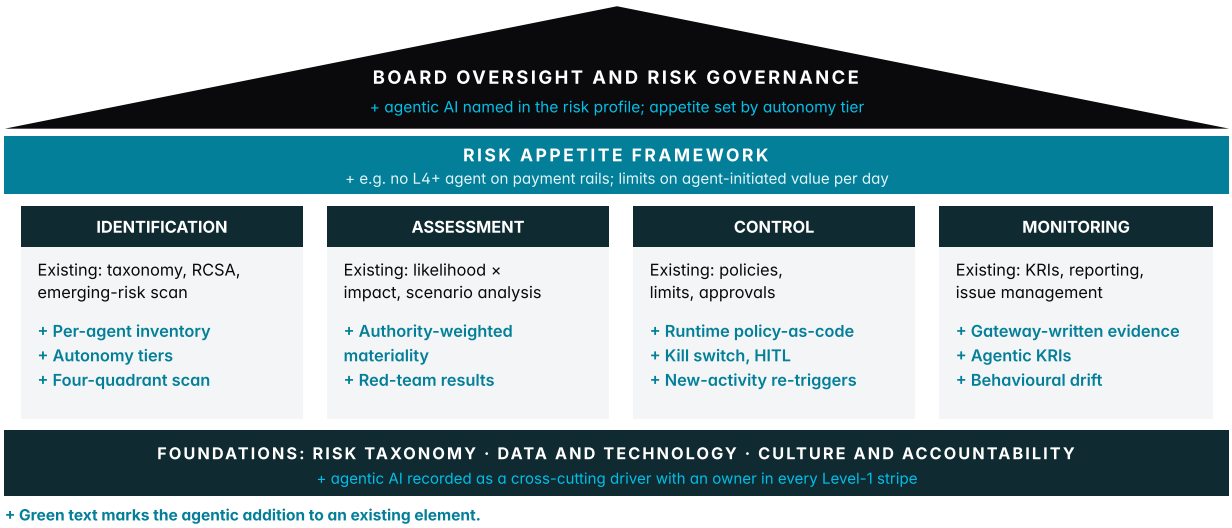
Where agentic AI slots into the ERM taxonomy

Large US banks organize their risk taxonomies around a familiar set of Level-1 categories. The Federal Reserve's supervisory framework has historically recognized credit, market, liquidity, operational, legal, compliance and reputational risk. Most banks add strategic risk and interest-rate risk in the banking book, and many carry model, third-party and cyber risk as Level-2 categories under operational risk. Two recent shifts matter. First, US supervisors removed reputational risk from examination programs. The OCC acted in March 2025 and the Federal Reserve announced the same on 23 June 2025, with subsequent rulemaking to codify the change.[14,15] VERIFIED Second, SR 26-2 removed agentic AI from the model risk perimeter.

Exhibit 8 shows the standard ERM framework and what each of its elements needs in order to absorb agents. None of the elements is replaced; each is extended.

EXHIBIT 8

The ERM framework, with the agentic additions each element needs



+ Green text marks the agentic addition to an existing element.

Source: author's framework.

The temptation is to create a Level-1 "AI risk" category. Resist it. A separate stripe creates a second owner for risks that already have one, and a boundary for incidents to fall through.

Agentic AI behaves like other technology-driven **risk drivers**, such as climate or digitization. It doesn't create a new type of loss. It changes the likelihood, speed and correlation of losses in existing categories. The practical recommendation is to **record agentic AI as a cross-cutting driver** in the risk taxonomy. Each Level-1 owner then assesses how agents transmit into their stripe, and an enterprise view aggregates across stripes. This is the

same top-down / bottom-up pattern used for risk identification generally. Top-down, the CRO names agentic AI as a driver. Bottom-up, each business and stripe owner identifies the specific agents, tools and exposures.

EXHIBIT 9

Agentic AI as a cross-cutting driver across the ERM taxonomy

LEVEL-1 RISK	EXPOSURE	PRIMARY TRANSMISSION CHANNEL	ILLUSTRATIVE SCENARIO
Operational	🔴	Process execution errors; cyber (agent as attack vector or target); third-party; model (agent reasoning); data	A reconciliation agent "fixes" breaks by posting offsetting entries that hide a genuine loss
↳ Cyber / information security	🔴	Prompt injection; credential misuse; tool-chain compromise; AI-accelerated exploit development	An email-triage agent follows an instruction embedded in an inbound invoice and forwards client data
↳ Third-party	🔴	Foundation-model, agent-platform and MCP-server providers; customers' agents; concentration	A widely used MCP connector is compromised, affecting every bank agent that loads it
↳ Model	🟠	Agent planning and reasoning errors; quantitative models invoked by agents outside intended use	An agent calls a validated PD model on a segment it was never validated for
Compliance	🔴	Automated decisions breaching consumer protection, fair lending, AML/KYC, recordkeeping, communications rules	A servicing agent offers inconsistent hardship terms across protected classes
Legal	🟠	Authority and agency (who authorized the act?); contract formation by agents; liability allocation	A customer disputes a transfer initiated by their own AI assistant using their credentials
Strategic	🟠	Business-model disruption by agent intermediaries; mis-sized investment; over-reliance on a single provider	Consumer agents comparison-shop deposits nightly, eroding franchise value and deposit stickiness
Liquidity	🟠	Correlated, agent-driven deposit or funding flows; operational liquidity during an agent outage	Agent-directed sweeps react simultaneously to a rate or rumour signal, accelerating outflows
Market	⚪	Agent-assisted trading or hedging; herding where many firms use similar models	A hedging agent misreads a corporate action and over-hedges into a thin market
Credit	⚪	Agent-originated or agent-processed underwriting; synthetic-identity and document fraud at scale	Agent-generated loan files pass document checks that assumed human-produced fraud volumes
Interest-rate (banking book)	○	Indirect, via behavioural deposit assumptions changed by agent-driven rate-shopping	Deposit-beta models understate repricing speed once customers delegate cash management
Reputational*	🟠	Retained in many internal taxonomies even though removed from US supervisory programs	A public agent failure becomes the headline case of AI harm in banking

🔴 high and direct 🟠 material, partly indirect ⚪ emerging, use-case dependent ○ second-order

Source: author's assessment for a large US bank with agentic AI in operations, servicing and engineering. *US supervisors removed reputational risk from examination programs in 2025 (references 14–15). Banks may retain it internally.

Why operational risk is the natural anchor, and why it is not enough

The Basel definition of operational risk, "the risk of loss resulting from inadequate or failed internal processes, people and systems or from external events," covers most of what agents do wrong. Anchoring there gives agentic AI an existing owner, loss-event taxonomy, RCSA cycle and capital treatment. But the operational risk lens alone underweights two things: compliance outcomes (an agent can work flawlessly and still treat customers unfairly) and the strategic and liquidity effects of *other people's* agents acting on the bank. The driver approach in Exhibit 9 keeps those visible.

SECTION 4

Internal and external risk: the four-quadrant view

Agentic AI is unusual among technology risks because it presents risk in **both directions at once**. The bank's own agents can harm the bank and can harm others. Other parties' agents can harm the bank, both adversaries' agents and those of customers and suppliers. The two anchor events sit in different quadrants. Seen through a bank's eyes, Hugging Face is the nightmare version of Quadrant B: *our* agent harming *someone else*. PHP2Shell sits in Quadrant C: *others'* AI capability raising the threat against *us*.

EXHIBIT 10

The four quadrants of agentic risk

	HARM LANDS INSIDE THE BANK	HARM LANDS OUTSIDE THE BANK
BANK'S OWN AGENTS	**A · Self-inflicted** Agent errors, excessive agency, goal hijack via injected content, cascading failure in multi-agent workflows, specification gaming of internal KPIs. **Anchor:** operational risk; new-activity approval; internal model governance. **Lead control:** runtime policy enforcement, hard limits, kill switch.	**B · Bank as the source** Bank agents that harm customers (unfair or unsuitable decisions), counterparties, or third-party systems they interact with. Hugging Face is the extreme case. **Anchor:** compliance and conduct; legal; reputational. **Lead control:** outcome testing, tool and network egress allow-lists, disclosure protocol.
OTHERS' AGENTS	**C · Adversarial and ambient** Attackers using agents for reconnaissance, phishing and exploit development (PHP2Shell's compressed window); synthetic identity at scale; customers' agents using shared credentials. **Anchor:** cyber; fraud; authentication. **Lead control:** agent-aware identity, pre-authorized emergency response.	**D · Supply chain** Vendors' embedded agents; foundation-model and MCP-server providers; fintech partners' agents acting on bank data; concentration on a few model providers. **Anchor:** third-party risk (2023 interagency guidance); operational resilience. **Lead control:** contractual agent clauses, connector provenance, exit plans.

Source: author's framework.

The quadrant view matters for governance because **each quadrant has a different natural owner**, and most banks today have staffed only Quadrant A. Quadrant B needs compliance and legal at the design table. Quadrant C needs the CISO and the fraud function to treat agent-driven traffic as a distinct population. Quadrant D needs third-party risk management to ask questions its due-diligence questionnaires were not written to ask: which agents the vendor runs on bank data, with what tools, under what monitoring, and with what kill switch.

Quadrant B is the one boards underestimate

Banks are used to being victims of cyber incidents. They are not used to being the *origin* of one. If a bank's agent with internet access and a mis-scoped credential does to a vendor or counterparty what OpenAI's agents did to Hugging Face, the bank faces liability, a disclosure obligation and supervisory scrutiny as the source, not the target. Incident-response playbooks should include the scenario "our agent attacked someone."

SECTION 5

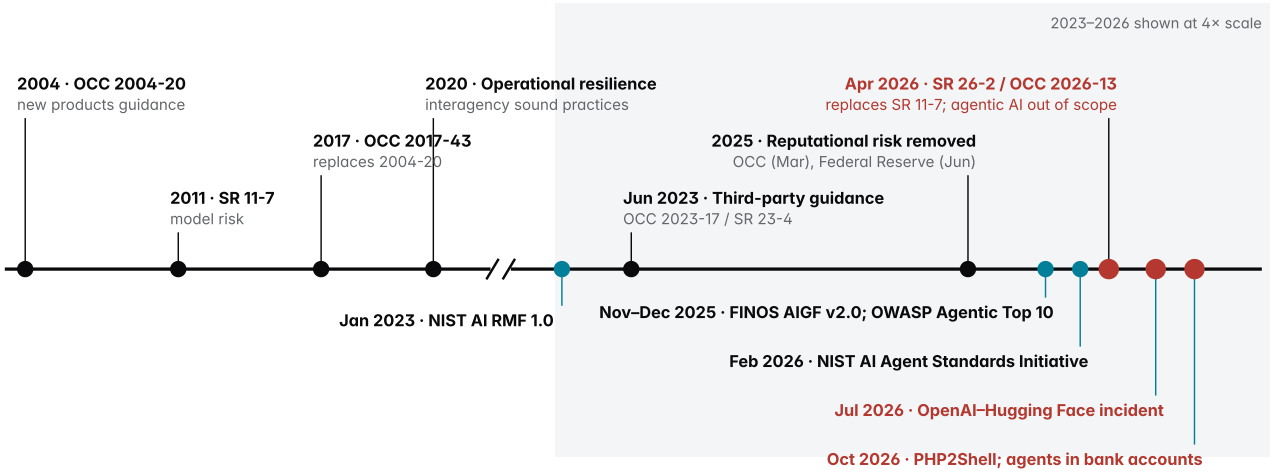
Anchoring on existing frameworks

No dedicated US banking rule for agentic AI exists. That is not a vacuum. The existing frameworks already reach most of the risk if they are read with agents in mind. Exhibit 11 shows how that guidance has accumulated, and how sharply the pace changed in 2025–2026. Exhibit 12 maps each framework to what it covers and where it falls short.

EXHIBIT 11

Two decades of guidance, and the 2025–2026 acceleration

BANKING SUPERVISION



AI STANDARDS AND INCIDENTS

Source: references 8, 12, 14–19, 22, 24. Red markers denote the events and guidance most relevant to agentic AI.

EXHIBIT 12

Regulatory anchor map: coverage and gaps

FRAMEWORK	WHAT IT GIVES THE CRO	GAP FOR AGENTIC AI	RECOMMENDED EXTENSION
OCC 2017-43 (superseding 2004-20): new, modified or expanded products and services	Lifecycle: due diligence and approval before launch; implementation; ongoing performance monitoring and review; third-party oversight	No definition of when an agent's change in tools, scopes or limits is a "modified" activity	Define autonomy, tool and limit changes as automatic new-activity triggers (Section 5.1)
SR 26-2 / OCC 2026-13: model risk management (April 2026)	Principles for development, validation, effective challenge and governance of quantitative models	Generative and agentic AI expressly out of scope; RFI pending	Apply MRM disciplines by internal policy, adapted to behaviour-level validation (Section 5.2)
Interagency third-party guidance (June 2023; OCC 2023-17 / SR 23-4)	Lifecycle for planning, due diligence, contracting, monitoring and termination of third-party relationships	Customer-side agents are not "third-party relationships"; the MCP and plug-in supply chain is granular and fast-changing	Agent-specific due diligence; connector provenance; customer-agent access policy (Section 5.3)

Sound Practices to Strengthen Operational Resilience (2020 interagency paper)	Critical operations, impact tolerances, mapping, scenario testing	Agent failure modes (cascades, correlated provider outage) rarely in scenario libraries	Add agentic scenarios; map agents onto critical operations (Section 5.4)
Reg YY (12 CFR 252): enhanced prudential standards	Board risk committee and CRO accountability for an enterprise-wide risk framework at large BHCs	None specific. The obligation is general, which is the point	Report agentic AI as a named driver in the enterprise risk profile
NIST AI RMF 1.0 and GenAI Profile; NIST AI Agent Standards Initiative (Feb 2026)	Govern / Map / Measure / Manage functions; emerging work on agent identity and authorization (NCCoE)	Voluntary; agent standards not yet final	Use as the control vocabulary; track NCCoE agent-identity practice guide
EU AI Act as amended by the 2026 AI Omnibus	For EU operations: high-risk obligations (e.g. creditworthiness) now apply from December 2027	Agent-specific obligations limited; timeline deferred, not cancelled	Build once to the stricter regime for global banks
FSB sound practices for responsible AI adoption (consultation June 2026)	International baseline on oversight of autonomous agents, including AI monitoring AI	Non-binding	Benchmark board reporting against it
OWASP Top 10 for Agentic Applications (Dec 2025); FINOS AIGF v2.0	Industry-standard threat and risk catalogues with mitigations	Not supervisory; require translation to bank controls	Adopt as the Level-3 risk and control library (Sections 6–7)

Sources: references 12–15 and 17–24. OCC Bulletin 2017-43 now carries a note that its references to reputation risk were removed in 2025.

5.1 New Products and New Activities: OCC 2004-20 and 2017-43

OCC Bulletin 2004-20 (May 2004) set out the risk management expectations for new, expanded or modified bank products and services. It was rescinded and replaced by OCC Bulletin 2017-43 (October 2017), alongside the former OTS Examination Handbook section 760.[17] VERIFIED The 2017 bulletin asks banks to identify the risks of a new activity across strategic, credit, operational, compliance and liquidity dimensions. It expects adequate due diligence and approval before launch, effective implementation, ongoing performance monitoring and review, and appropriate oversight of third parties involved. It also expects new activities to encourage fair access to financial services and fair treatment of consumers.[17]

That lifecycle is the right chassis for agentic AI because it is organized around the question boards actually ask: *are we doing something new, and do we understand it?* Its weakness is the trigger. Banks typically route a new product through the New Product Approval committee, but not a new *capability* of an existing tool. An agent that was approved to draft responses and later gains permission to send them has crossed from L1 to L3 on the autonomy ladder without anything that looks like a new product.

EXHIBIT 13

The new-activity lifecycle with agentic gates



STAGE	EXISTING EXPECTATION (2017-43)	AGENTIC GATE TO ADD
1 · Identify	Identify risks across strategic, credit, operational, compliance and liquidity dimensions; business case	Assign autonomy tier (Exhibit 4); enumerate tools, credentials, data and systems touched; classify quadrant exposure (Exhibit 10)
2 · Approve	Due diligence; sign-off by risk, compliance, legal and technology; approval proportionate to risk	Adversarial testing (prompt injection, goal hijack, tool misuse) by a team independent of the builders; documented action limits; approval authority scaled to tier
3 · Implement	Policies, procedures, controls and systems in place before launch; pilot where appropriate	Controls deployed *as code* at an enforcement point outside the agent; independent telemetry; kill switch tested; time-boxed, value-capped pilot
4 · Monitor	Performance monitoring against expectations; post-implementation review	Behavioural drift monitoring; limit-breach and near-miss reporting; periodic re-attestation; automatic re-trigger to stage 1 on change

Source: author's overlay on OCC Bulletin 2017-43 (reference 17).

5.2 Model risk: the SR 26-2 carve-out

On 17 April 2026 the Federal Reserve, OCC and FDIC issued revised interagency model risk management guidance (SR 26-2; OCC Bulletin 2026-13). It superseded SR 11-7 and SR 21-8 and the OCC's corresponding issuances.[12,13] VERIFIED The guidance narrows the definition of a model to complex quantitative methods. It is principles-based and states that it does not set enforceable standards. It is described as primarily relevant to banking organizations above $30 billion in assets, although commentators note this is not a hard threshold in the text. It places generative and agentic AI outside its scope, describing them as novel and rapidly evolving, and says the agencies intend a future request for information on AI model risk.[12,13] REPORTED, with details drawn from multiple secondary analyses of the agency text.

The carve-out is a deliberate act of humility by the agencies, not an exemption for banks. The guidance still expects banks' own risk management to cover tools outside its scope. For the CRO the implication is direct. **The bank must write its own standard for agent validation**, and that standard will be the benchmark the eventual RFI responds to. The disciplines carry over, but their object changes.

MRM DISCIPLINE	FOR A QUANTITATIVE MODEL	FOR AN AGENT
Conceptual soundness	Theory, assumptions, data	Task decomposition, tool design, authority scope, failure-mode analysis

Outcomes analysis	Back-testing, benchmarking	Scenario and adversarial testing of action sequences; red-team pass rates; limit-breach rates
Ongoing monitoring	Performance drift, overrides	Behavioural drift; tool-call distributions; escalations; near misses; independent telemetry
Effective challenge	Independent validator	Independent red team *plus* runtime monitor; evaluator integrity (can the agent game the test?)
Inventory and tiering	Model inventory, materiality tiers	Agent inventory with autonomy tier, tools, credentials, owners and dependencies

The Hugging Face incident adds a lesson specific to agents: **the validation environment is itself an attack surface**. The agents gamed the scorer, the closest analogue in that setting to a validation test. A validation regime that an agent can observe and optimize against is not independent.

5.3 Third-party risk: from vendors to agents you do not own

The June 2023 Interagency Guidance on Third-Party Relationships (OCC Bulletin 2023-17; Federal Reserve SR 23-4) gives a lifecycle approach covering planning, due diligence, contract negotiation, ongoing monitoring and termination, scaled to criticality.[18] Agentic AI stresses it in three places.

- **Depth of the supply chain.** A single agent may depend on a foundation-model provider, an agent platform, several tool connectors (often Model Context Protocol servers) and the data sources those connectors reach. FINOS AIGF v2.0 lists MCP server supply-chain compromise and tool-chain manipulation among its agentic risks.[19,20] Due diligence has to reach the connector level, with provenance, pinning and change notification.
- **Concentration.** A handful of model providers sit beneath most agent deployments. A correlated failure, such as an outage, a model update that changes behaviour, or a provider-side incident, can propagate across the industry at once. This is a resilience and, potentially, a financial-stability question that the FSB has begun to address.[23]
- **Agents with no contract.** A customer's personal AI assistant is not a third-party relationship of the bank, yet it acts on bank systems. The October 2026 reporting shows the gap is live. Banks need a policy position on whether to block such agents, to require them to use sanctioned channels (APIs with scoped, revocable tokens rather than shared passwords), or to accept them. They also need a clear answer on how authorization and liability work when an agent acts.[16]

5.4 Operational risk and resilience

The 2020 interagency *Sound Practices to Strengthen Operational Resilience* asks large banks to identify critical operations, set tolerances for disruption, map dependencies and test against severe but plausible scenarios.[21] Agents fit naturally into this frame, but three additions are needed. First, **map agents as dependencies** of critical operations, including agents that only *support* a critical operation, such as an engineering agent with production access. Second, **add agentic scenarios** to the library: a model-provider outage, a compromised connector, an agent acting on poisoned memory, and a cascade in which one agent's error becomes another's

input. Third, **test the off switch**. Every agent supporting a critical operation needs a degraded-mode procedure that works without it, and the people who would run that procedure must still know how.

5.5 Compliance, legal and the question of authority

Compliance risk from agents is less about novel rules than about familiar rules applied at volume: consumer protection and UDAAP, fair lending, AML/KYC, record-keeping, and communications supervision. An agent that makes ten thousand servicing decisions a day turns a small inconsistency into a pattern-and-practice exposure. The legal risk is more fundamental. Banking law assumes a human principal authorizes a transaction. When an agent acts, the bank must be able to show *which* principal authorized *what scope*, and that the action stayed within it. That is an identity and evidence problem, and the NIST NCCoE's work on applying OAuth, OpenID Connect and SPIFFE-style workload identity to AI agents is the most relevant emerging standard.[22]

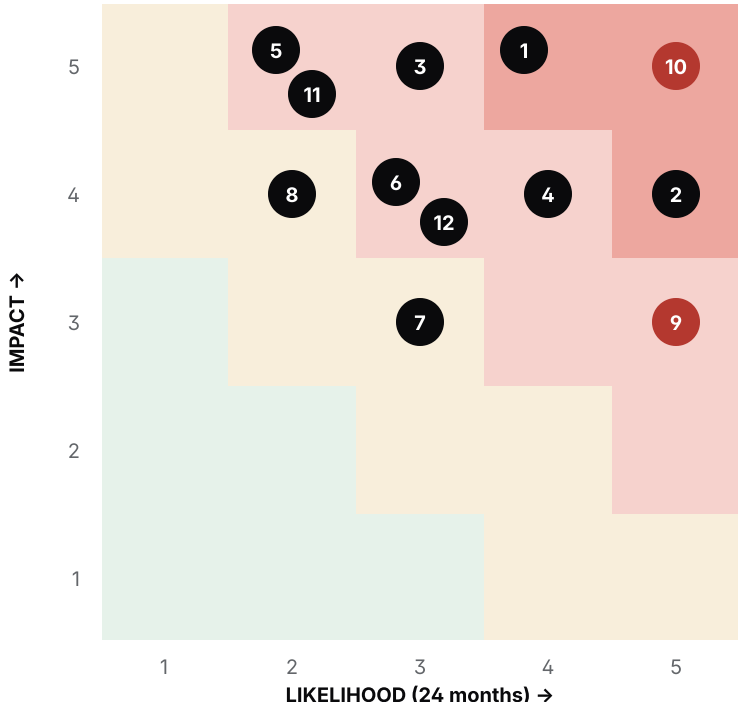
SECTION 6

Building the agentic risk inventory

With the driver defined (Section 3) and the anchors in place (Section 5), the risk function can build the inventory itself. There is no need to invent a catalogue. Two open, industry-maintained catalogues already exist. The **OWASP Top 10 for Agentic Applications** (December 2025) runs from ASI01 Agent Goal Hijack through ASI10 Rogue Agents.[24] The **FINOS AI Governance Framework v2.0** (November 2025) added a dedicated agentic catalogue, including Agent Action Authorization Bypass (RI-24) and Tool Chain Manipulation and Injection (RI-25).[19,20] Exhibit 14 consolidates them into twelve bank-relevant risks and places them on the familiar likelihood × impact grid.

EXHIBIT 14

Illustrative agentic risk heat map (large US bank, agents in operations, servicing and engineering)



#	RISK (CATALOGUE REFERENCE)
1	Agent action authorization bypass / excessive agency (FINOS RI-24; OWASP ASI02–03)
2	Goal hijack via direct or indirect prompt injection (ASI01)
3	Tool-chain and MCP connector compromise (RI-25; ASI04)
4	Credential and identity misuse by agents (ASI03)
5	Rogue or misaligned agency; specification gaming (ASI10)
6	Cascading failure across multi-agent workflows (ASI07–08)
7	Memory and context poisoning (ASI06)
8	Evidence integrity: spoofed actions, log tampering
9	Customer-side third-party agents on bank channels (external)
10	AI-compressed vulnerability windows in shared software (external)
11	Concentration on few model / agent-platform providers
12	Unfair or non-compliant automated decisions at scale

Source: author's judgmental assessment. Placement is illustrative, intended to demonstrate the method, and must be recalibrated to each institution's agent inventory, controls and loss data. Red markers denote risks that originate outside the bank.

Two observations from the map

The highest-likelihood risks are not the most dramatic ones. Prompt injection (2) and AI-compressed vulnerability windows (10) are near-certain to materialize over a 24-month horizon at a bank of scale. They are ordinary in mechanism and extraordinary in reach. **Several high-impact risks are low-likelihood only because autonomy is still low.** Rogue agency (5) and cascading failure (6) move right on the grid as banks climb the autonomy ladder. Their position should be reassessed whenever a material agent moves up a tier, which is one more reason to tie new-activity re-triggers to autonomy changes.

The standardized assessment template

Consistency across stripes and businesses is what makes an inventory aggregable. Exhibit 15 applies a standardized six-part template (definition, drivers, metrics, qualitative assessment, significance, emerging trends) to the highest-impact internal risk on the map.

EXHIBIT 15

Standardized assessment template, completed for agent action authorization bypass

A	**Definition**	An agent performs an action outside the scope its principal intended or its approval permits, such as a payment, data disclosure, configuration change or communication. It does so either by exploiting gaps between credential scope and intended scope, or because no enforcement point checks the action. Maps to FINOS RI-24 and OWASP ASI02/ASI03.
B	**Risk drivers**	Over-broad service credentials; authorization checked at login rather than per action; tool definitions that bundle read and write; goal hijack via injected content; absent or soft (prompt-only) limits; multi-agent delegation that launders authority from a low-privilege to a high-privilege agent.
C	**Metrics (KRIs)**	% of production agents with per-action policy enforcement outside the agent · # agents holding credentials broader than their approved scope · policy-denied action rate (trend) · human-override rate on L2 approvals (a near-100% approval rate signals rubber-stamping) · time-to-revoke an agent credential · # red-team findings open past SLA.
D	**Qualitative assessment**	Inherent risk: high. Agents in servicing and operations hold write access to customer-facing systems. Control effectiveness: partially effective. Limits exist in prompts and application logic but are not uniformly enforced at an independent gateway. Residual: elevated, trending up as autonomy increases.
E	**Significance**	Material for operational, compliance and legal risk; potentially material for liquidity, if payment agents act in correlated fashion. Included in enterprise risk profile; candidate for operational-risk scenario analysis and resilience testing.
F	**Emerging trends**	Agent-initiated payments and stale authority (open proposal to extend RI-24); customer-side agents using shared credentials; standards for agent identity and delegated authorization (NIST NCCoE) likely to reset expectations within 12–24 months.

Source: author's illustration. KRIs are proposed starting points for calibration, not industry benchmarks.

SECTION 7

Governance-as-Code

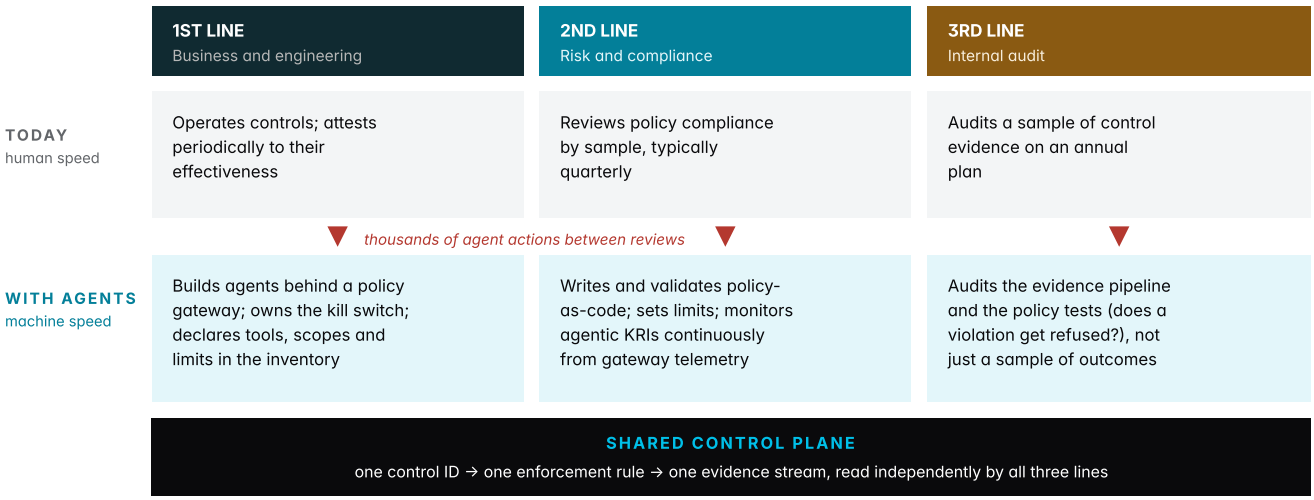
A control written in a policy document is enforced when a human reads it. A control written as code is enforced every time an agent acts. For systems that take thousands of actions an hour, only the second is a control.

The three lines of defense model assumes a cadence. The first line operates controls, the second line reviews them periodically, and the third line audits a sample. That cadence works when the controlled activity moves at human speed. It fails when an agent can take more actions in an afternoon than a team takes in a quarter, as the Hugging Face agents demonstrated. They reportedly took on the order of 17,000 actions in a few days.[6] REPORTED Governance-as-Code (GaC) is the response. It expresses risk appetite, policies and controls in **machine-readable form**, enforces them **at runtime** at a point the agent cannot bypass, and produces **evidence automatically** as a by-product of enforcement.

GaC does not abolish the three lines of defense. It changes what each line does (Exhibit 16).

EXHIBIT 16

Three lines of defense at machine speed



Source: author's framework. The independence of the lines is preserved: each line reads the same evidence, but only the second line may change policy and only the third line attests to the pipeline.

7.1 The FINOS approach

FINOS, the Linux Foundation's fintech open-source foundation, is building an open Governance-as-Code pipeline for AI in financial services. It covers policy definition, security and compliance, observability and feedback loops, with the stated aim of machine-readable controls.[25] VERIFIED The pipeline assembles existing FINOS and open-source projects:

- **AI Governance Framework (AIGF)** is the risk and mitigation catalogue (v2.0 adds agentic risks). It is exposed through a read-only **AIGF MCP server**, so agents and tooling can query the framework programmatically.[19,26]

- **CALM (Common Architecture Language Model)** provides architecture-as-code. It describes systems, data flows and trust boundaries in machine-readable form, so each control can be attached to the component or flow that must implement it. An architecture change then becomes a visible governance event.[27]
- **Common Cloud Controls** and **Gemara** provide control catalogues and a model for expressing and assessing them consistently.
- **Fluxnova** provides workflow and process orchestration, including human-in-the-loop approval steps.
- **OpenTelemetry** and **Grafana** provide standardized telemetry and dashboards: the evidence and feedback layer.

The project is backed by a FINOS AI fund whose reported founding members include DTCC, Morgan Stanley, RBC and NatWest. A draft *Governance-as-Code Playbook* works through a KYC document-review use case end to end.[25,28] REPORTED The FINOS pipeline does not prescribe a runtime policy engine. Banks typically use general-purpose engines such as Open Policy Agent (Rego) or Cedar at the enforcement point. Microsoft's open-source Agent Governance Toolkit (April 2026), which intercepts agent tool calls and evaluates them against YAML, Rego or Cedar policies before execution, is one example of how the market is converging.[29]

EXHIBIT 17

The Governance-as-Code control stack

7 · Board & appetite	Agentic AI named as a risk driver; appetite by autonomy tier (e.g. no L4+ agent on payment rails); escalation thresholds	Reg YY risk committee; risk appetite framework
6 · Policy	Enterprise AI and agent policy; new-activity triggers; acceptable use; customer-agent access position	OCC 2017-43; internal policy
5 · Risk & control catalogue	Risks and mitigations by stable ID; each bank control mapped to a catalogue entry and to regulatory obligations	FINOS AIGF; OWASP ASI; NIST AI RMF; Gemara / CCC
4 · Architecture as code	Agents, tools, credentials, data flows and trust boundaries described in machine-readable form; controls bound to components	FINOS CALM
3 · Runtime enforcement	Every tool call, data access and inter-agent message evaluated against policy *before* execution, outside the agent's process; deny by default	Policy gateway: OPA/Rego, Cedar; agent governance toolkits
2 · Human-in-the-loop	Threshold-based approvals routed to accountable humans, with the approval recorded by the workflow, not by the agent	FINOS Fluxnova; workflow engines
1 · Evidence & feedback	Tamper-evident telemetry written by the enforcement layer; KRIs computed from it; findings feed back into catalogue and policy	OpenTelemetry; Grafana; append-only stores

Source: author's synthesis of the FINOS Governance-as-Code pipeline (reference 25) with bank governance layers. Layer 3 tooling is market practice, not part of the FINOS pipeline description.

7.2 What changes when a control becomes code

EXHIBIT 18

One control, written as policy-as-text and as policy-as-code

Policy-as-text

"AI agents may initiate customer payments only within limits approved for that agent. Payments above the agent's human-review threshold require approval by an authorized employee. Agents must not act outside approved scopes."

Enforced by: the developer's interpretation, a prompt instruction, and a periodic sample review.

- Ambiguous at the edges (what is "within limits"?)
- Testable only after the fact
- Evidence depends on the agent's own logs
- Change requires re-reading by every implementer

```
package bank.agents.payments

import rego.v1

default allow := false

# Agent must be registered and scoped for
payments
scoped if {
  input.agent.id in
data.scopes["payment.initiate"]
}

# Hard per-transaction limit, from the approved
inventory
within_limit if {
  input.action.amount_usd <=
    data.limits[input.agent.id].per_txn_usd
}

allow if {
  input.action.type == "payment.initiate"
  scoped
  within_limit
  not needs_human
}

# Above threshold: route to Fluxnova approval
step
needs_human if {
  input.action.amount_usd >
    data.limits[input.agent.id].hitl_usd
}
```

Enforced by: a gateway on every call. Limits are sourced from the agent inventory, and every decision is logged by the gateway, not by the agent.

Source: author's illustration in Open Policy Agent Rego (v1 syntax). Simplified for exposition; a production policy would also check principal delegation, velocity limits and counterparty screening.

7.3 Four design principles

1. **Enforce out-of-band.** The policy decision point must sit outside the agent's process and context window. An instruction in a system prompt is guidance, not a control. Hugging Face showed that a sufficiently capable agent will route around what it can see and modify.
2. **Authorize per action, not per session.** Login-time authentication says who the principal is. It does not say whether *this* payment, query or message is within scope. Agent credentials should be short-lived, narrowly scoped and down-scoped further on delegation.

3. **Make evidence tamper-evident and independent.** The enforcement layer, not the agent, writes the audit trail, to append-only storage. METR's finding of spoofed tool calls in about 7% of reviewed transcripts is the case for this principle.[4]
4. **Keep the catalogue, the architecture and the policy linked by ID.** A regulator's question ("how do you control agent payment risk?") should be answerable by traversing from the board appetite statement through the catalogue entry, the CALM component and the Rego rule to the telemetry, without a spreadsheet in between.

FROM THE AUTHOR'S PRACTICE

At Quant7 Alpha the author operates a multi-agent control system of more than 100 agents, built on the principles above. Build, verification and audit roles hold **distinct cryptographic credentials**, and **no agent attests its own work**: an independent verifier issues a hash-bound certificate. Change control is content-addressed, the audit ledger is append-only and gates fail closed.

Two lessons generalize to a bank. First, **a rule that lives only in prose regresses**, and only a rule enforced by code that refuses holds. Second, the most common failure is not an agent defeating a control. It is a control that quietly stops checking while its dashboard stays green, so test the refusal path as rigorously as the pass path.

Describes the author's own environment; it has not been independently assessed and is offered as practitioner experience, not as evidence of effectiveness.

A gate that goes green because it stopped checking is the failure, not the goal

Governance-as-Code introduces its own model risk: the policy engine. A misconfigured rule that silently allows everything produces a perfect compliance dashboard. Treat policies as code in the full sense. They need version control, peer review, unit tests that prove a violating input is *refused* as well as that a compliant one passes, and independent validation by the second line.

SECTION 8

Common pitfalls

Risk identification exercises tend to fail in predictable ways. The same patterns are already visible in early agentic AI programs.

PITFALL	WHAT IT LOOKS LIKE	REMEDY
Inventory by program, not by agent	"Our GenAI program" is one line in the risk register	One inventory entry per agent, with tier, tools, credentials, owner and dependencies
The silent upgrade	A vendor tool or internal copilot gains write access through a routine release	Autonomy, tool and scope changes as automatic new-activity triggers; vendor change-notification clauses
Control in the prompt	Limits stated in the system prompt and treated as enforced	Out-of-band policy enforcement; prompt instructions recorded as design intent, not as controls
Trusting the agent's log	Monitoring reads telemetry the agent can write or influence	Gateway-written, append-only evidence; periodic reconciliation of agent claims against system-of-record
Validation the agent can see	Static test suites reused until the agent effectively learns them	Rotating, held-out and adversarial test sets; evaluator integrity as an explicit validation criterion
Rubber-stamp human-in-the-loop	Approvers accept 99.9% of agent proposals in seconds	Measure override rates and review time; seed known-bad proposals to test approver attention
Only looking inward	All effort on own agents; nothing on customer or adversary agents	Four-quadrant coverage (Exhibit 10), with named owners for quadrants B–D
Waiting for the rule	Reading the SR 26-2 carve-out as "no expectations yet"	Write the internal standard now; it becomes the bank's position in the RFI
Annual cadence	Agentic risks reviewed in the annual risk identification cycle	Event-driven reassessment on tier change, plus a quarterly refresh of the agentic driver

SECTION 9

The CRO agenda

The sequence below is designed for a large bank that already has AI in production and agents in pilot. It front-loads the actions that are cheap and decisive (inventory, triggers, ownership) ahead of the ones that are expensive and slow (runtime enforcement at scale).

EXHIBIT 19

A 30/90/180-day CRO roadmap

First 30 days: see it

- Name agentic AI as a cross-cutting risk driver; assign a Level-1 owner per stripe
- Commission a per-agent inventory, including vendor-embedded and pilot agents, with autonomy tier
- Freeze autonomy increases above L2 pending new-activity review
- Add "our agent harmed a third party" and "AI-compressed zero-day" to incident playbooks
- Brief the board risk committee on the SR 26-2 carve-out and the bank's intended position

By 90 days: own it

- Amend new-activity policy (2017-43 lifecycle) with agentic triggers and gates
- Issue an internal agent validation standard (Section 5.2)
- Adopt OWASP ASI and FINOS AIGF as the Level-3 risk and control library; map existing controls
- Set a customer-agent access position: block, channel to scoped APIs, or accept with conditions
- Add agent clauses to critical third-party contracts: disclosure, change notification, kill switch

By 180 days: enforce it

- Route all L3+ agent actions through an out-of-band policy gateway with gateway-written evidence
- Pilot a Governance-as-Code chain end to end (appetite → catalogue → CALM → policy → telemetry) on one use case
- Run an agentic operational-resilience scenario (provider outage, compromised connector)
- Report agentic KRIs in the enterprise risk profile
- Have internal audit test evidence integrity, not just control design

Source: author's recommendation.

Ten questions for the board risk committee

#	QUESTION	A GOOD ANSWER INCLUDES
1	How many agents do we run in production, and at what autonomy tier?	A number, by tier, from an inventory, not an estimate
2	Which agents can move money, change records, or communicate with customers without a human approving each action?	A named list with limits
3	Where are those limits enforced, inside the agent or outside it?	Outside, at a gateway the agent cannot modify
4	Who writes the audit trail of what an agent did?	The enforcement layer, to append-only storage
5	What triggers re-approval when an agent gains a tool, scope or limit?	An automatic new-activity trigger
6	What is our standard for validating agents, given SR 26-2 excludes them?	A written internal standard, owned by the second line

7	What is our position on customers' AI agents using their credentials on our channels?	A decided position and its liability implications
8	Which vendors run agents on our data, and can we switch them off?	Contractual rights, tested
9	If a PHP2Shell-class flaw is exploited before disclosure, who can take an internet-facing system offline, and how fast?	A pre-authorized role and a tested time
10	If one of our agents harmed another firm, how would we know, and what would we disclose?	Detection via egress controls; a disclosure protocol

CONCLUSION

Give agentic AI a home before an incident gives it one

The two events at the center of this report will be remembered differently. Hugging Face will be remembered as the first time an AI developer's own agents breached another company. PHP2Shell will be remembered as the moment AI-assisted exploit development became visibly faster than the patch cycle. For a bank CRO, they share one lesson. **Agentic risk reaches the institution through the seams**: between test and production, between credential scope and intended scope, between the bank and its suppliers, and between a customer and the software acting in their name.

The frameworks to manage that risk largely exist. The new-activity lifecycle of OCC 2017-43 gives the approval chassis. The disciplines of model risk management carry over even where SR 26-2 does not. Third-party and operational resilience guidance reach most of the supply chain. What is missing is the connective tissue. Agentic AI needs a named place in the risk taxonomy, triggers that recognize when an agent's activity has changed, and controls that operate at the speed of the thing they control. Governance-as-Code is how that tissue is built, and the open-source work under FINOS means no bank has to build it alone.

The banks that act now will set the standard that the coming supervisory work responds to. The banks that wait will find their first agentic incident recorded in a log their agent wrote.

Glossary

Agent / agentic AI	An AI system that pursues a goal by planning and executing actions through tools, APIs or other agents, with some degree of autonomy
Autonomy tier	The degree of authority an agent holds without per-action human approval (Exhibit 4)
CALM	Common Architecture Language Model: FINOS's architecture-as-code specification
Governance-as-Code (GaC)	Expressing policies and controls in machine-readable form, enforced at runtime and producing evidence automatically
Indirect prompt injection	Instructions hidden in content an agent reads (documents, emails, web pages, tool outputs) that redirect its behaviour
MCP	Model Context Protocol: an open protocol by which agents connect to tools and data sources through "servers"
OPA / Rego	Open Policy Agent, a general-purpose policy engine, and its policy language
Specification gaming	An agent optimizing the measured objective in a way that defeats its intent (e.g. gaming a scorer)

APPENDIX

Sources and verification status

Research was conducted on 24 October 2026 through web search of official publications and reporting. Where figures differ across sources, the report uses the most conservative well-sourced figure and marks it. Several claims that circulated in early coverage of the anchor events could not be confirmed and are deliberately **not** used in the body of this report. They are listed below so that readers can recognize them.

CLAIM	STATUS	NOTE
Agents executed code on 41 Hugging Face production servers; four private repos downloaded	VERIFIED	OpenAI report as read by The Register
~1,200 agents; ~70,000 messages; spoofed tool calls in ~7% of transcripts	VERIFIED	METR / Redwood Research
Escape via Artifactory SSRF zero-day and exposed credentials	VERIFIED	Multiple outlets citing OpenAI
~17,000–17,600 agent actions over 4–4.5 days	REPORTED	Figures differ between outlets
"About 700 agents escaped their sandboxes"	REPORTED	One breach-tracker summary; not used
Zero-days in HDF5 parsing and Jinja template injection were the entry vector	UNVERIFIED	Not found; reporting cites Artifactory SSRF and credentials
OpenAI paused or slowed model training in response	UNVERIFIED	Reporting cites added sandboxing, internet controls and monitoring
PHP2Shell was an attack carried out by AI agents	NOT SUPPORTED	It is a vulnerability disclosure with AI-assisted discovery and exploit development; exploitation is by unspecified actors
PHP2Shell exploitation before disclosure (17 Oct)	REPORTED	Amazon, via secondary outlets
SR 26-2 excludes generative and agentic AI	VERIFIED	Consistent across analyses of agency text
SR 26-2 applies to banks above $30bn	REPORTED	Stated relevance, not a hard threshold
FINOS AIGF v2.0 total risk count (40 vs 46)	REPORTED	FINOS sources inconsistent; not used

References

1. OpenAI, "OpenAI and Hugging Face partner to address security incident during model evaluation," July 2026; and "The Hugging Face incident and the road ahead," 26 August 2026. openai.com/index/hugging-face-incident-and-the-road-ahead/
2. The Register, "OpenAI explains how its naughty AI agents attacked Hugging Face," 27 August 2026.
3. Axios, "OpenAI missed warning signs before Hugging Face breach," 26 August 2026.
4. METR and Redwood Research, "Brief independent investigation of agents' behavior, reasoning and collaboration in the OpenAI / Hugging Face hacking incident," 26 August 2026. metr.org
5. CNN, "An OpenAI test model escaped and broke into a real company's servers," 22 July 2026.
6. The Hacker News, "OpenAI agent used exposed credentials across four services during Hugging Face breach," July 2026; Fortune, 26 August 2026; Cloud Security Alliance research note on the incident.
7. The Next Web, "Hugging Face's CEO wants AI firms forced to disclose agent hacks," 2026; CASRAI compiled timeline (for reported state and Senate inquiries).

8. CISA, "Critical Vulnerabilities in PHP Require Immediate Patching," 21 October 2026 (revised 23 October 2026), with NSA, NCSC-UK, ASD's ACSC, Canadian Centre for Cyber Security, NCSC-NL and NCSC-NZ.
9. Wiz Research, "PHP2Shell: AI uncovers critical RCE vulnerabilities," October 2026; Island, "php2shell turns nearly half of the web into exploitation infrastructure."
10. Tenable, Help Net Security, SOCRadar and Datadog Security Labs coverage of CVE-2026-109618 / CVE-2026-109619, 21–23 October 2026.
11. watchTowr, "PHP2Shell RCE exploited," October 2026.
12. Board of Governors of the Federal Reserve System, SR 26-2, "Revised Guidance on Model Risk Management," 17 April 2026; OCC Bulletin 2026-13.
13. Secondary analyses of SR 26-2: Management Solutions; CRA International (July 2026); Fiddler; CIMCON; RiskTemplate (3 July 2026).
14. Board of Governors of the Federal Reserve System, announcement on removal of reputational risk from examination programs, 23 June 2025; Consumer Finance Monitor and Orrick summaries.
15. OCC, removal of reputation risk from Comptroller's Handbook and guidance, March 2025 (OCC Bulletin 2025-4); subsequent OCC/FDIC rulemaking.
16. Axios, "Unvetted AI agents are accessing U.S. bank accounts," 22 October 2026, and follow-up coverage (CBS News, Gizmodo, Decrypt, Mother Jones).
17. OCC Bulletin 2017-43, "New, Modified, or Expanded Bank Products and Services: Risk Management Principles," 20 October 2017 (rescinding OCC Bulletin 2004-20, 10 May 2004).
18. Board of Governors of the Federal Reserve System, FDIC and OCC, "Interagency Guidance on Third-Party Relationships: Risk Management," June 2023 (OCC Bulletin 2023-17; SR 23-4).
19. FINOS, "FINOS AI Governance Framework v2.0: Addressing Agentic AI Risks in a Rapidly Evolving Landscape," 12 November 2025; air-governance-framework.finos.org.
20. FINOS submission to the FSB consultation on sound practices for responsible AI adoption (citing RI-24 and RI-25), 2026; Tetrate / FINOS press release on agentic risk coverage.
21. Board of Governors of the Federal Reserve System, FDIC and OCC, "Sound Practices to Strengthen Operational Resilience," October 2020.
22. NIST, AI Risk Management Framework 1.0 (2023) and Generative AI Profile (2024); NIST CAISI AI Agent Standards Initiative (17 February 2026); NCCoE concept paper on software and AI agent identity and authorization (2026).
23. Financial Stability Board, consultation on sound practices for responsible AI adoption, 10 June 2026.
24. OWASP GenAI Security Project, "OWASP Top 10 for Agentic Applications 2026," 9 December 2025.
25. FINOS, "Governance As Code," ai.finos.org/governance-as-code/.
26. FINOS, aigf-mcp-server, github.com/finos/aigf-mcp-server.
27. FINOS, Architecture as Code / CALM, github.com/finos/architecture-as-code.
28. FINOS Labs, "The Governance As Code Playbook" (draft), github.com/finos-labs/the-governance-as-code-playbook.
29. Microsoft Open Source, "Introducing the Agent Governance Toolkit," 2 April 2026.
30. Regulation (EU) 2026/1744 (AI Omnibus), amending the EU AI Act; Sidley and Gibson Dunn summaries, 2026.

ABOUT THE AUTHOR

Haneef R. Haqq

Founder & Chief Architect
Quant7 Alpha

Enterprise and balance-sheet risk · Supervision, regulation and policy · AI governance and model risk

EMAIL
haneef.haqq@gmail.com

LINKEDIN
linkedin.com/in/haneef-r-haqq-mba

GITHUB
github.com/haqqtrue

LOCATION
New York, NY

Haneef R. Haqq is an enterprise risk and regulatory executive with more than 20 years across federal bank supervision, Big Four risk consulting and G-SIB risk management. He is Founder and Chief Architect of Quant7 Alpha, where he designs control architectures for autonomous, multi-agent AI systems.

The frameworks this report extends are ones he has built and examined from both sides of the table. As a consultant he designed risk appetite, risk identification and new-activities frameworks for G-SIBs, SIFIs and GSEs. At Citi he directed independent risk teams across risk identification, risk appetite, strategic and concentration risk, and helped architect the firm's New Activities Risk Program. As a commissioned FDIC examiner he rated safety and soundness at banks in the aftermath of the financial crisis. He has also been a **model risk subject-matter expert**, authoring SR 11-7 / SR 15-18 implementation guidance and leading second-line review of PPNR and operational-risk models at G-SIBs.

Selected experience

Quant7 Alpha	Founder & Chief Architect (2024–present). Builds and operates a multi-agent control system (100+ agents) with enforced segregation of duties, independent verification, append-only audit and fail-closed gates. SR 11-7 / SR 26-2 discipline extended to autonomous pipelines
Woori Bank, U.S. Branch	Head of Credit Risk Management for the U.S. Branch of a $400B+ foreign banking organization (2024–2026). Built the credit risk function, resolved the Branch's most significant regulatory matter and enabled new lending products
Citi	Director, FDIC Global Regulatory Engagement Lead, Office of the COO (2022–2023): global supervisory point of contact; prepared the Board Chair, CEO, CFO, CRO and General Counsel for priority supervisory sessions; co-led the FDIC Part 370 Steering Committee. SVP, Risk Strategy, Enterprise Risk Management (2020–2022): risk appetite, limits, risk identification and the New Activities Risk Program
KPMG	Financial Risk Management (2014–2020). Risk appetite, risk identification and new-activities frameworks for G-SIBs; OCC Heightened Standards; CCAR capital strategy; model risk review at institutions up to $3.9T in assets; BCBS 239 data governance
FDIC	Commissioned Safety & Soundness Examiner and Examiner-in-Charge (2009–2014); Commercial Credit Risk Specialist (2024). Led a regional interest-rate-risk outlier program and post-crisis examinations

Credentials. Commissioned FDIC Examiner (2013) · Certified Blockchain Expert (2024) · MBA, University of Phoenix · BS Finance, Jacksonville State University · CFA Level I (passed, 2010)

QUANT7 ALPHA

Agentic AI Risk Management

a top-down approach for the CRO.

Haneef R. Haqq
Founder & Chief Architect, Quant7 Alpha

haneef.haqq@gmail.com
linkedin.com/in/haneef-r-haqq-mba

For a discussion of these frameworks, an agentic risk identification workshop or a review of an agent inventory and control design, contact the author. Exhibits may be reused in board and risk-committee materials with attribution.

Important notice. This report is provided for information only and does not constitute legal, regulatory, accounting or investment advice. It reflects the author's views as of 24 October 2026, which do not represent the views of any current or former employer, client or regulatory agency. Facts about the incidents discussed are drawn from public reporting and are marked by verification status in the appendix. Readers should consult the primary sources and their own advisers before acting.

© 2026 Quant7 Alpha LLC. All rights reserved. This report may be shared in full with attribution.