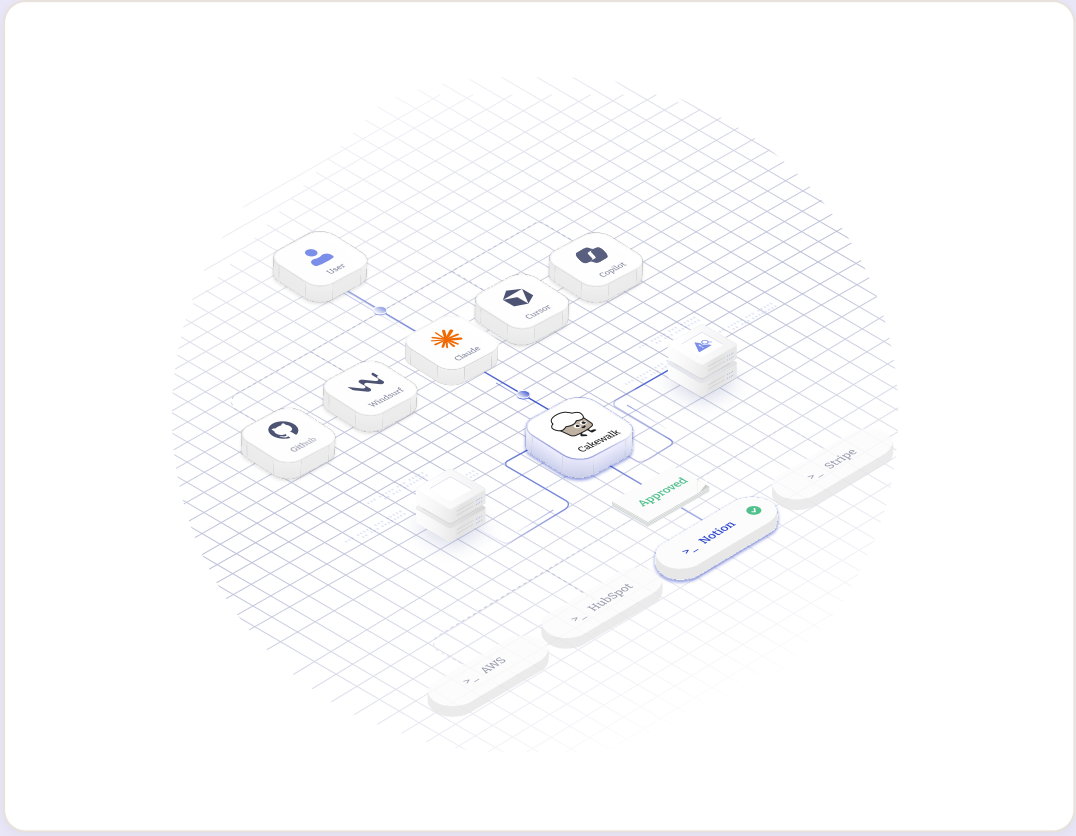


Human → Agent → System: A Working Model for Governing Agent Access

How a line on our 2023 roadmap became security's biggest unsolved problem and led to the access layer we are now building.

DATE 2026.05.05

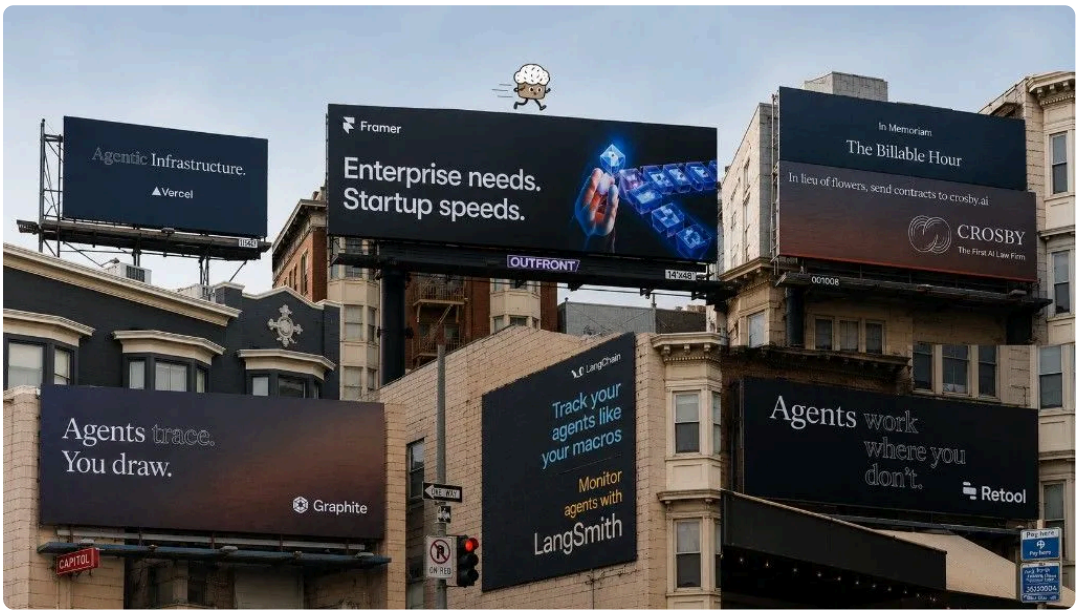
AUTHOR JOHANNES KEIENBURG, CEO & FOUNDER

READING TIME 8 MIN READ

CATEGORIES STORY

San Francisco, May 2026: AI Agents Will Be the Biggest Revolution in Human History. But...

If you want to understand to what extent the world is changing these days, you have to look at the billboards in San Francisco: AI agents everywhere. Agents for sales, agents for engineers, agents for everything. Every ad, every meetup, every coffee chat: same theme.



Autonomous agents will be the biggest revolution in humankind. To make this revolution work and go the right way (something that comes up in every conversation), we have to get the security fundamentals right. And one of the biggest unsolved security challenges for autonomous agents is how to govern agent access.

Why? Because without enough agent access governance, breaches are already happening.

The Vercel Breach: From a Compromised Laptop to a $2 Million Sale of Stolen Data

On April 19, Vercel disclosed a breach. The chain began months earlier. Context.ai is an enterprise AI platform with a Google Workspace integration. Its AI Office Suite builds agents trained on company-specific knowledge. A Vercel employee signed

up for it using their Google Workspace account and granted Context.ai "Allow All" OAuth permissions. Researchers later traced the chain back to February, when an infostealer compromised a Context.ai employee's laptop and opened a path into the OAuth grants Context.ai held for its customers.[2] The attacker used the Vercel grant to walk into Vercel's Workspace, then into the internal environment. They exfiltrated what Vercel called "non-sensitive" environment variables: API keys, tokens, database credentials and signing keys.[1] The attacker listed the data for sale on the criminal forum BreachForums for $2 million.[2] Vercel's CEO described the attackers as "highly sophisticated and, I strongly suspect, significantly accelerated by AI."[4]

> Guillermo Rausch @rauchg
>
> Here's my update to the broader community about the ongoing incident investigation. I want to give you the rundown of the situation directly.
>
> A Vercel employee got compromised via the breach of an AI platform customer called Context.ai that he was using. The details are being fully investigated. …
>
> 2026-04-21 READ THE FULL POST ON X →

The breach moved through four links: a compromised vendor laptop, a vendor with stored OAuth grants, a Vercel-issued credential and an enterprise Workspace. None of them had been reviewed in living memory or required anyone to be in the loop at the moment they were used.

The layer that would govern this kind of chain is still scattered across the industry.

The Vercel Pattern Repeats Across the Industry

Across these breaches, the same shape returns. The Salesloft Drift breach in August 2025 reached more than 700 organizations, including Cloudflare, Zscaler and Palo Alto Networks.[3] Context.ai is the latest in the line.

The mechanics are familiar by now: OAuth grants that accumulate faster than anyone reviews them, consent screens that bundle every future call into a single yes, audit trails that name the human at signup but not the agent at runtime. The trust model under OAuth was built for a single-vendor application acting on a user's behalf. Once the grant is given, almost nothing watches what the vendor does with it.

In two earlier pieces, we mapped why agentic access is the new frontier in security and the current protocols and enforcement architectures to tackle this. Every solution we surveyed closes one edge of the chain, sometimes two, while none closes all three. That gap is the one we keep running into. The shape it leaves exposed is the one we have been preparing Cakewalk to govern: *Human → Agent → System*.

The Three-Actor Topology: From Machine-to-Machine to Human → Agent → System

When we started Cakewalk back in 2023, our mission was to build Access Management that works across the entire workforce: simple enough for any employee to use and yet powerful enough for IT and security teams to rely on.

Early on, our roadmap included what we called "machine-to-machine access" back then: one corner of what the industry calls non-human identity (NHI). The phrase fit a small and well-behaved set of workloads at the time: a handful of service accounts, a few CI tokens, the occasional cron job that needed to talk to a database without a person in the loop. For those workloads, identity took the form of an OAuth grant or an API key issued to a piece of software, with scope set at deploy and behaviour written into the code ahead of time. Two actors and a credential between them: a Machine → System model that worked because the credential's scope was small and predictable.

By 2026, AI agents inherited the same credentials as machine-to-machine access but with none of the predictability. The reason is simple, in that traditional non-human identities do not act autonomously, while autonomy is exactly what makes access governance crucial. An agent therefore needs runtime primitives similar to those a human needs: who delegated to it, which system it is acting against and with what authority.

The new shape that emerged with agents is a triangle with three actors at its corners: a delegating human, an acting agent and a target system. Every agent inherits its right to act from the human who delegated to it. Every action that agent takes is directed at the system with its own constraints. Every access decision is now defined across all three positions at once. Each major failure of the past year sits on one of these three edges of the triangle.

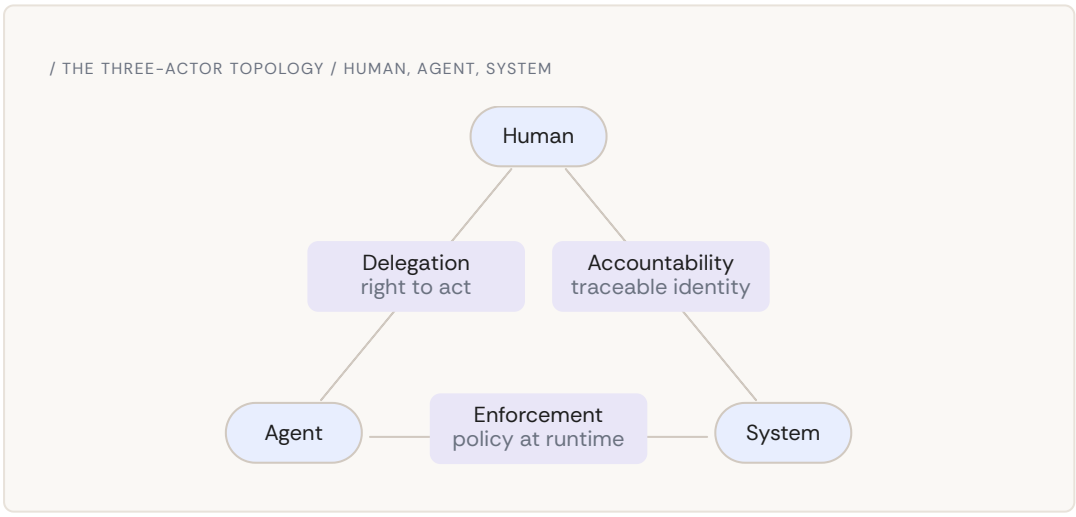
Accountability rests on the human edge. Without the human kept in the chain at the moment of access, accountability disappears. Every action becomes anonymous, attached to a credential rather than a person. Compliance frameworks that

assume a human at the end of the chain (e.g., SOC 2, NIS2) stop working in any auditable sense. So does anyone trying to reconstruct what happened after the fact.

The Vercel shape comes from the agent edge of the triangle. When the agent itself is invisible to the layer that would govern it, what is left between the human and the system is a static OAuth grant exercised at the agent's discretion, with no enforcement in the path. The grant might have been reasonable on the day it was issued. Whether it is still reasonable today, used by an agent with new capabilities and inside a context the original grant did not anticipate, is a question the layer is not equipped to answer.

The system edge is the one most often dropped from the conversation, on the assumption that the agent's context already matches the system it is calling. There is no way to confirm that the agent's context matches reality. The same agent action that is safe against a sandbox is destructive against production. Without the target system in the picture, no policy can be evaluated against what the call will do. The fact that the call was made is all that registers.

The breaches we studied this year, including Vercel and Salesloft Drift, fit this picture. Each one is a triangle with one edge missing.



How We Are Building Agent Access Management

We are building the access layer that closes all three sides at once. We call it Agent Access Management at Cakewalk. The difference from any earlier sketch is that the human is now inside the access decision, not just in the audit trail.

The credential is where the Vercel chain breaks down first. In Cakewalk's gateway, every token exists for the duration of one session: permissions are injected the first time the agent needs them and revoked the moment the session ends. Cakewalk keeps the tokens in its own vault, separate from the agent. The agent's context window is its most exploitable surface, because agents are built to read untrusted input and a single prompt injection can pull anything stored there out into the open. Keeping the tokens in a separate vault means even a compromised agent has no tokens to leak. One gateway sits between the agent and every system it reaches, replacing dozens of unmanaged grants and removing the Agent → System surface the Salesloft/Context.ai pattern depends on.

What happens at each call is the next decision. Where the OAuth model presents the user with one decision at signup ("Allow") that covers every subsequent call the application makes, our gateway evaluates each individual tool call as it is made. The gateway reads three inputs at every evaluation: the action being requested, the delegating human and the target system. The result is a deterministic, reproducible policy decision specific to that combination, meaning the same input produces the same answer and that answer can be replayed at any later time. All three edges of the triangle are closed at the same evaluation, because it is the only point in the chain where all three are visible at once.

The delegating human is part of every decision, not just the audit trail. Every action an agent takes is tied back, in the audit trail and in the policy decision itself, to the human who delegated to it. When that human's role or department changes, every agent acting on their behalf inherits the change instantly. The audit trail is built around the delegating human first and the credential second. That inverts OAuth's static grant model and makes "who owns this agent" answerable at runtime. The Human → Agent accountability that disappears under the static-grant model is restored.

From the Defining Question to the Central Layer We Are Building

Three years ago we put machine-to-machine access on the Cakewalk roadmap. That same line is now the defining question in enterprise security and the Human → Agent → System layer is what we are building to answer it. The companies that deploy agents safely over the next several years will

treat agent access as infrastructure, the way they already treat identity and networking.

And that is what's needed to turn billboard talk into global reality, securely.

/ REFERENCES

[1] Vercel (2026). April 2026 Security Incident. vercel.com

[2] Tom's Hardware (2026). Vercel breached after employee grants AI tool unrestricted access to Google Workspace. tomshardware.com

[3] Google Cloud / Mandiant (2025). Data theft from Salesforce instances via Salesloft Drift. cloud.google.com

[4] Rauch (2026). Statement on Vercel security incident. x.com