

Blog 5: Who Is This AI Agent, Anyway? The Identity Problem

From the CSA Agentic AI Summit Series

By TekSol Global | Cybersecurity & AI Practice

Here's a question that sounds simple but turns out to be surprisingly deep:

When an AI agent does something inside your company's systems - who is it acting as?

Is it a machine? Is it acting on behalf of a human? Is it its own thing? Should it have its own login? Its own permissions? Its own accountability trail?

These questions sit at the heart of one of the biggest security challenges in AI today: **identity**.

Why Identity Matters in Security

In the physical world, identity is how we control who gets access to what.

You have a key to your house, but not your neighbour's. Your employee badge works for the floors you need, not the executive suite. Your bank card has your name on it, and the bank knows exactly which transactions are yours.

Identity creates **accountability** (we know who did what) and **access control** (we can decide who's allowed to do what).

In the digital world, the same principle applies. Every system, every service, every person needs a known identity so that:

1. You can verify it's really them
2. You can limit what they're allowed to do
3. You can trace their actions if something goes wrong

Now, where does an AI agent fit into all of this?

The Survey That Revealed a Mess

At the summit, the Cloud Security Alliance presented research from a survey of over 200 IT and security professionals, asking exactly how their organizations are managing AI agent identity.

The results painted a messy picture.

- Only **36%** of organizations had given AI agents their own dedicated identity
- **52%** were treating agents like regular applications (which doesn't quite fit)
- **43%** were using shared or generic service accounts (which means multiple agents, or even agents and humans, sharing the same login)
- Many were just letting agents borrow the identity of whatever human user had set them up

And here's the consequence: **68% of organizations said they couldn't clearly tell the difference between actions taken by AI agents and actions taken by humans** in their own systems.

Think about what that means. If an agent does something problematic, or if a hacker hijacks an agent, you can't even tell it happened because you can't distinguish the agent's activity from normal human activity.

The "Who Did This?" Problem

Let's make this concrete with a simple example.

A company has an AI agent that helps manage its file storage. One day, a large number of confidential documents are moved to an external location.

The security team investigates. They look at the logs. They see: *action performed by "admin_user_3."*

But who, or what, is admin_user_3? Is it an actual employee? Is it the AI agent that was set up by that employee and inherited their login? Was the employee's account compromised by a hacker, who then used it to control the agent?

Without a clear AI agent identity, you can't answer these questions. You can't know if this was:

- An employee making a mistake
- An agent behaving unexpectedly
- A hacker who got into the system
- A compromised agent is doing something it shouldn't

The investigation goes in circles. The trail goes cold.

Old Identity Systems Weren't Built for This

The tools companies use to manage identity, known as **IAM systems** (Identity and Access Management), were designed for humans and traditional software.

A human logs in once. They do their work. They log out. Simple.

AI agents are different. They:

- Can run continuously for days without stopping
- Can spin up dozens of "sub-agents" to handle different tasks
- Can switch contexts rapidly, acting on behalf of one user in one moment, another in the next
- Can operate across multiple systems simultaneously
- Can make thousands of access requests per hour

Traditional identity systems weren't designed for this kind of behaviour. An agent making 10,000 database queries in one hour would look like a cyberattack to most traditional monitoring systems, but it might just be the agent doing its job.

The Token Problem

In tech, a **token** is basically a temporary pass. Instead of using a password every single time a system needs to verify identity, you get a token after logging in, and that token says "yes, this is a verified user" for a set period of time.

AI agents rely heavily on tokens to access systems. Here's where things get dangerous:

Long-lived tokens: Many agents are set up with tokens that never expire, or that last for months. If a hacker steals one of these tokens, they have persistent access to whatever that token allows, indefinitely.

Over-privileged tokens: Because developers want agents to be flexible, they often give them tokens with very broad permissions. A token that grants access to everything is a master key, and master keys are very attractive targets.

Token sprawl: As agents are connected to more services, they accumulate more tokens. Keeping track of all of them, who has them, what they allow, and whether they're still needed, becomes nearly impossible without dedicated tools.

One speaker described this as walking around with every key to every door in a building hanging from your belt, jangling around for anyone to grab.

What Good AI Identity Looks Like

The summit speakers described what a properly secured AI agent identity looks like. It's not complicated in concept; it's just not widely implemented yet.

Every agent should have its own unique identity. Not shared with humans, not shared with other agents. It's own, distinct, trackable identity.

Agents should have only the access they need. If an agent's job is to summarize documents, it shouldn't have access to the payroll database. This principle is called **least privilege**.

Access should be temporary and specific. Instead of a permanent pass, agents should get short-lived tokens for specific tasks, and those tokens should expire as soon as the task is done.

Every action should be logged. Every system an agent touches, every document it reads, every action it takes, all of it should be recorded in a way that's searchable and auditable.

Agents should be traceable to a human owner. Someone in the company should be responsible for each agent, the person who authorized its deployment and whose job it is to ensure it behaves properly.

Why This Is Harder Than It Sounds

Here's the frustrating reality: even companies that understand all of this are struggling to implement it.

The tools to give AI agents proper identities are still being built. The standards for how it should work are still being written. The industry is essentially trying to retrofit a security principle (identity management) onto a technology (agentic AI) that moves faster than the security frameworks can be designed.

One summit speaker put it bluntly: *"The existing primitives for dealing with identity are not entirely suited for tracking agent identities."*

That's a polite way of saying: the old tools don't work for this new problem, and we're building the new tools in real time.

The Good News

The conversation at the summit was not just about problems. It was about solutions.

Organizations like Okta are specifically working on identity systems designed for AI agents. The principles are clear, even if implementation is still catching up. And the fact that the security community is focused on this problem now, before AI agents are deployed at full scale everywhere, is a good sign.

Understanding that AI needs its own identity is step one. Building the systems to enforce it is step two. And as we'll see in the next blog, identity is just one piece of a larger framework for keeping AI agents under control.

Next up → Blog 6: The Five Questions Every AI Agent Must Answer Before You Trust It