Re(AI)magine Conversations

Persistent Podcast | Episode 16

The New Data Security Reality in the Age of AI

Akshay Chitlangia and Simon Thornell

Akshay Chitlangia: Hi everyone. Welcome to Re(AI)magine Conversations, where we explore how AI technology and bold thinking are transforming the enterprises of tomorrow. I'm your host Akshay Chitlangia, VP Technology at Persistent Systems. Joining me today is Simon Thornell, Field CTO at TrustLogix.

Simon, why don't you introduce yourself?

Simon Thornell: Yeah, it's great to meet you actually. So my name's Simon Thornell, Field CTO at TrustLogix. Kind of being very much focused around access, control and authorization with AgTech (Agricultural Technology) deployments out there in the environments. But prior to that, I was kind of very much focused around things like encryption within RAG pipelines, confidential computing infrastructure for AI models.

So, hopefully have some good experience that I can help share today.

Akshay Chitlangia: Thanks, Simon. So, in today's episode, we'll explore how enterprises can enable secure, trusted data access for their AI agents, why governance often lags AI velocity at enterprise scale and what it really takes to make agent AI systems work safely at scale.

So, let's dive into this topic now. AI adoption across every enterprise function is accelerating. However, governance is not keeping pace. We're seeing clients deploy autonomous agents faster than they can secure them. That's why today's conversation is so important, not just for us, but for our customers as well.

Simon, from your vantage point at TrustLogix, what's the first thing enterprises get wrong when agents enter the picture?

Simon Thornell: Yeah. I think to me, you know, data access governance is falling behind and that's very much caused by the AI velocity gap. What I mean by that really is kind of, AI is operating at machine speed, whereas your kind of legacy controls and logging don't essentially.

So, things like standing privileges, manual approvals, you know, they're going to break instantly when we have autonomous agents starting to use things. Like a PP tool requests access to various data platforms, and really what that does is it kind of gives organizations a bit of a false choice. You know, do they slowdown that AI adoption, or do they start to accept unmanaged risk?

And, you know, neither are going to work. Neither pass the test. There's additional complexity that starts to get added in as well. You know, things like, Sovereign AI that adds a completely different dimension. But it's the same underlying theme really.

As enterprises start to deploy these AI agents globally, the data residency jurisdictional controls become mandatory. They were there before AI, they're still there with AI. So, you have to kind of conform with that. And you know, really, agents must start to respect where data can flow as well as who can access it, why and how.

This is something we've been very much working on with enterprises. With our TrustLogix TrustAI platform, we've been trying to address this directly, providing things like policy-driven authorization controls, visibility for agents, MCB-based workflows, so that data access governance kind of keeps pace with the speed of the agentic AI deployment.

Akshay Chitlangia: Exactly. We are now seeing this in a lot of real customer implementations. Let's talk about the hidden risks of AI super users.

When an AI agent holds a very broad service account credential, I want to treat them as a super user with excessive reach. With MCP sprawl, this becomes worse because the surface area, from a risk point of view, connected to AI agents is every connector, every tool server, every gateway. One incorrect prompt chain can touch entire data sets far beyond what a human role should have ever been able to access.

Unfortunately, clients tend to underestimate how quickly the blast radios of AI agents expands. Simon, what do you see as the biggest identity and privileged blind spots right now with your clients?

Simon Thornell: The best example of it is the one that you gave. You know that that agent having a service account. That has a greater privilege of the user. And I think one of the biggest blind spots we start to see is enterprises lose track of who, whether that's human or agent, actually access the data and what did they, what did they do and what they were they entitled to do. But there are also bigger blind spots that we start to think about as well, because those are just focused on identity, where we aren't doing anything around like full agent attribution evaluation either. So, we may well have things like the user identity, we may well have the agent identity, although that is still a big problem for people.

But do we have things like the declared purpose of the agent? So, when a response is issued, can we check that against the declared purpose of the agent to make sure the response aligns with it? And then things like sensitivity checks, do they need to apply? And you know, without things like least privilege, just-in-time access, these agent purpose and attribute checks against agents, they're very much becoming like unbounded.

And that's where we see the huge blind spots to start to emerge. That's where we've been focusing on shrinking that blast radius with our clients. But if we think about other blind spots as well, you know, we've spoken about, okay, there's that control layer and that access blind spot. But there's also that logging layer as well. You know, specifically for agent interactions, we're finding that to be a huge blind spot. Have you got the logs? Can you identify risks in those logs? So not just focusing on the end data source, but then also like, what else is the agent using MCP servers, tool usage?

We really need to start looking into that as well. So, I think some of the key points we've picked up on to kind of help close off the identity and privileged blind spots is, you know, everybody we're working with, it's the same common theme.

Policies must evaluate the context of every single request. Who is making the call for what data, for what purpose, under what conditions and why. And that all needs to be logged because really these static and allow and deny policies that we typically had from a security perspective, they're very much dead. In this case they need to be intelligent. They need to be adaptive permissions.

That's very much the future and what you need for agentic AI. you know, if they're running within the same infrastructure as the end data source. Perhaps we can deploy native policies that integrate into those target systems and those controls can run at that same speed as AI because it's designed to the platform it's running in.

But then if we start to think about agents residing either on other frameworks within data sources or outside of these data sources, then we need to start looking at things like MCP, which has huge benefits when it comes to user and agent authorization and how we can actually do this logging, how can we actually do this authorization regardless really of where the agent or the, the model is.

What you can't do is just solely focus on that single access point. What we need to start doing is shift left and start looking at user-agent framework. The interaction between the agent and the end data source. We need to be picking up across that whole layer so that we move away from, from those blind spots.

You know, one of the questions that clients ask me all the time, they always go, look, this is very good. Completely understand it, but how do we implement this without slowing down the delivery?

Akshay Chitlangia: Honestly, that's a very fair question from both you and clients. Nobody wants governance to become a blocker that kills momentum.

Ultimately, the business has to work. The business has to move ahead. The way we've seen it work is when governance becomes part of the engineering ecosystem and not an extra approval layer. So, instead of adding new committees, you build the right kind of platform controls. You have the right policy code mechanisms. You have the right reusable patterns for identity propagation.

You have the appropriate standard guardrails for tool access, for tool control, and you have the audit telemetry, the logs that you've been referring to. Again, the context cannot just mean "who." It should also include where, and it should expand further in terms of what constraints apply to those context and to those access.

From a serenity point of view, we therefore help teams design it so that the agent can operate at speed inside the permitted boundary. What is allowed in a region, what can go across regions? What can operate in an efficient manner without breaking the compliance norms? The real unlock is progressive rollout.

Let's look at one more topic, bridging the human and non-human identities. A huge risk emerges when the agent's service identity is broader than the human's identity. We touched upon this a few minutes ago as well. If you don't propagate the requesting human through the AI workflow, the agents, in our opinion, can over access on their behalf.

If you look at it from this perspective, it might break segregation of duties, and it might also be foul of regulatory controls. Simon, one thing that keeps coming up for us in conversations is identity mismatch of this sort. The agent acts with broader permissions than the human who requested it. And then you've got the MCP sprawl that we've both referred to a few times already.

Tools, multiplying endpoints and APIs everywhere. How does TrustLogix help with this at runtime and keep everything controlled without slowing teams down? Could you tell us a little bit on this topic?

Simon Thornell: Yeah. I think that's a common problem that we hear. You know, there's always another agent or a user interacting with an agent.

And, we've come across all different use cases where, you know, user context is left lost, or it goes into a role. For example, agent context is lost and you basically can't pin back. You know, who did what and, and why did they do it. So, what we actually do is we bind the agent action against the actual human or other agent's identity, using attribute and purpose-based access as well.

So, if the human or the agent is entitled to or isn't entitled to the data or the MCP tool, then the actual agent that's being used won't see or use it either. And this can work the other way around as well. So, you know, the agent may carry access restrictions beyond what the human holds in either direction.

And really TrustLogix is enforcing the least amount of privilege possible from an entitlement perspective. Whether that's at the user level, the agent, the agent's framework, the data source, the MCP server, really trying to maintain that true lease privilege. And I think like the best way to kind of think about it is a common term.

We've heard a lot throughout security throughout the years. You know, this. TrustLogix's TrustAI, multi-layer entitlement check brings that defense in-depth. You know, we've heard a lot about defense in-depth in all kinds of security areas, but really we're doing that to ensure that neither the human nor the agent can exploit the other's broader permissions really.

So, privilege is never accidentally inherited in either direction, and we can log those interactions as well because that's critically important. You know, logging and continuous visibility, audit trails, they're kind of becoming more and more prevalent for multiple reasons really. And one of the reasons is we see that AI agents are often black boxes when they go into production or when they're trying to get them into production.

You know, if something goes wrong you can't tell which MCP tool or which request caused that data exposure. It's really, really tricky. So, we are logging every decision deterministically as well, and creating that order-to-ready evidence. You know, actually one of the things that I'd really like to understand as well is that, you know, this visibility layer that we are looking at, how does that then help the downstream enterprise functions?

Akshay Chitlangia: Well, visibility is what makes governance non-theoretical. It makes the governance aspects operationally useful once you have the logs that you referred to of what the agent requested, what it touched, what decisions it took, and why that becomes like a shared language across multiple teams, whether it is security or regulatory security.

Security teams can actually monitor the agent behavior using logs like they monitor any other high-risk workload or device. Audit and compliance teams get the evidence that they require so that they can have the transparency or explainability for specific activities or actions. Data governance teams, of course, get the answer to questions about which data is being used for which AI outcomes and under what constraints from which geography.

So, this telemetry or these logs can also now plug into GRC workflows. Next generation SOC monitoring and risk dashboards. This also starts to make incident responses faster because the teams are no longer guessing. They can trace the value chain by looking at the prom. The tools called the authorizations, the decisions made, and then the data access is done on the basis of those decisions culturally.

It also changes the conversation in the enterprises instead of saying, AI is a black box that we cannot trust. It allows the organizations to start saying, AI is governed in our organization like any other enterprise grade system. And if they can do that, that's when they can actually reach scale with AI and not be stuck or limited at pilots.

Now from a Persistent point of view, what we are really helping clients do is build the right kind of foundations to help them move agent care into production safely. In the real world, agents are never just add-ins or add-ons. There are actually identity propagation like we've talked about today, end-to-end.

It also is about aligning the right data accesses with the privacy InfoSec and governance policies, the right kind of observability. That, however, works at machine speed and making sure that the operating models are ready, not just the tech stack. So, we have to sit with the engineering, security, data governance and the business teams of our client and design all of this in a platform with the correct reusable patterns, scalable controls and a delivery approach that is aligned with their roadmaps.

Ultimately, the goal is simple AI that's usable and fast, but also compliant, auditable and defendable.

Simon Thornell: Here at TrustLogix, we are really kind of fortunate to work with partners like yourself at Persistent. And be part of that kind of end-to-end architecture that you help customers with.

So really from our perspective, TrustLogix within that architecture that you guys work with various enterprises on is we're providing that real-time authorization layer that kind of evaluates every query, every agent access requests and hopefully every bit of sensitive data that those agents are trying to touch.

Basically as part of a creating response, there's a couple of core themes that are part of that as well, so we've heard a lot about them today where we have dynamic context, aware entitlements, access policies for users, for agents. You know, that could stray into things like just-in-time access as well.

But these deploy policies are deployed across user agent, agent framework, MCP tools, clouds on-prem databases. Really, we're trying to become that unified control layer within that architecture. That also builds in things like the visibility, the audit trails, etc. And you know, I think the best way to kind of map it back to some industry terms that we're hearing is things like AI security, posture management, DSPM, continuous monitoring for AI.

But I think there's a lot of value with yourselves and with TrustLogix around what are the business benefits we deliver together as well. So, you know, we're starting to kind of shift that authorization and audit logging to TrustLogix, TrustAI platform. As part of an architecture and really what that helps enterprises do you hear quite, you know, terms like policy as code and we're moving from code to an abstraction layer.

And really what we're trying to do is take those tool access and permissions that are no longer hard coded into the agent logic shift those to that TrustLogix abstraction layer. And you know, some of the business benefits that come from that are things like. Even just dramatically shortening time to production for an agent, right?

How do you want to realize that return on investment quicker? So, if you think what we're enabling by doing that abstraction layer is things like fewer code reviews, no hardcoded permission changes, faster approval cycles, you know, because that governance layer is now outside of the agent, but then also like if you're a dev, the one thing you don't want to be doing is kind of playing around and having to code in your authorization piece. It's complex. You have to have multiple stakeholders, multiple insights. It's going to change. So, what you don't want that to do is become static in code. You want to allow your devs to focus on the functionality of an agent and start to deliver that use case, deliver that return-on-investment on the use case. And you know, there's a continued knock-on effect that comes from this.

So, we've realized the security benefits, but again, further business benefits are things like, think about the lifecycle of an agent. You know, you help enterprises set up that, that end-to-end architecture that you, you then work with them iteratively through that agent's lifecycle where we're going to come across things like new data sources, permission changes, changes that are driven by things like regulatory compliance that we spoke about today.

Where you are not recoding the agent every time and having to go back through all this review cycle, but you're using an abstraction layer that can dynamically update the agent. You know, there's a huge, huge business benefit there, as well as a security benefit. That's where we're trying to kind of position ourselves within that, that architecture that you are helping enterprises with.

Akshay Chitlangia: Alright, so we've established today that clients don't need just another tool for governance. They are looking for something they can operate an architecture of an operating model that holds up when they go. From two use cases to a hundred use cases on the agent AI side.

Simon Thornell: Yeah. and as part of that, you know, it was that kind of use case and the iterative use cases that come, they need that runtime intelligence that adapts to AI speed without slowing down that innovation as well.

Because you want to get it into production, you want to start to realize the benefits.

Akshay Chitlangia: So in summary, if you need AI systems, AI agents that you're piloting, you're looking to scale them up, you need to keep identity runtime, access and observability in mind. Talk to us, see how we can help you scale things up, how we can help you build the best that you can for your enterprise.

Simon Thornell: Yeah, I think you guys at Persistent are very much bringing in, you know, the expertise, the engineering, the knowledge from various kind of AI deployments.

And hopefully, we're bringing the technology, which means we can start to move these agent use cases into production and navigate that kind of complex regulatory and compliance frameworks that we have to adhere to as part of it.

Akshay Chitlangia: Okay. Thanks, Simon, for joining us today. I mean, it was interesting listening to your thoughts as the field CTO of TrustLogix and what you feel about AI scaling up in today's enterprises, especially around identity sprawl and data issues.

Simon Thornell: Yeah, as always. Akshay, it was brilliant to talk to you.

It's great to work with the team at Persistent and you know, hopefully we can carry on doing that for many years to come. I think one last little bit of advice to people is, don't wait for that breach or order issue to happen. Reach out and engage with partners like Persistent and vendors like TrustLogix and let's put that policy fabric in place early.

But yeah, it was great to talk to you again.

Akshay Chitlangia: Alright, so thanks for tuning in to Re(AI)magine Conversations. If today's episode got your interest, or you found something of interest, follow the show and share it with your network and with your colleagues. If you have a story to tell or a guest you'd love to hear from along with Persistent, drop us a note at podcast@persistent.com.

Until next time, stay curious, stay inspired. See you.

Re(AI)magining™ the World

About Persistent

Persistent Systems (BSE: 533179 and NSE: PERSISTENT) is a global services and solutions company delivering AI-led, platform-driven Digital Engineering and Enterprise Modernization to businesses across industries. With over 27,500 employees located in 18 countries, the Company is committed to innovation and client success. Persistent offers a comprehensive suite of services, including software engineering, product development, data and analytics, CX transformation, cloud computing, and intelligent automation. The Company is part of the MSCI India Index and is included in key indices of the National Stock Exchange of India, including the Nifty Midcap 50, Nifty IT, and Nifty MidCap Liquid 15, as well as several on the BSE such as the S&P BSE 100 and S&P BSE SENSEX Next 50. Persistent is also a constituent of the Dow Jones Sustainability World Index. The Company has achieved carbon neutrality, reinforcing its commitment to sustainability and responsible business practices. Persistent has also been named one of America's Greatest Workplaces for Inclusion & Diversity 2025 by Newsweek and Plant A Insights Group. As a participant of the United Nations Global Compact, the Company is committed to aligning strategies and operations with universal principles on human rights, labor, environment, and anti-corruption, as well as take actions that advance societal goals. With 468% growth in brand value since 2020, Persistent is the fastest-growing IT services brand in 'Brand Finance India 100' 2025 Report.

USA
Persistent Systems, Inc.
2055 Laurelwood Road, Suite 210
Santa Clara, CA 95054
Tel: +1 (408) 216 7010
Fax: +1 (408) 451 9177
Email: info@persistent.com

India
Persistent Systems Limited
Bhageerath, 402
Senapati Bapat Road
Pune 411016
Tel: +91 (20) 6703 0000
Fax: +91 (20) 6703 0008

Persistent
Re(AI)magining the World

www.persistent.com