Geordie

Executive Edition

The CISOs Guide to Agentic Security & Governance

Securing, governing, and scaling AI agents

JANUARY 2026

Table of Contents

Executive Summary

Across conversations with CISOs in 2025, a clear need has emerged for a grounded understanding of AI agents and what they mean for enterprise security. Many leaders are already seeing early examples of agentic behaviour across their organisations, yet the language and frameworks available today often fall short of describing how these systems think, decide, and act.

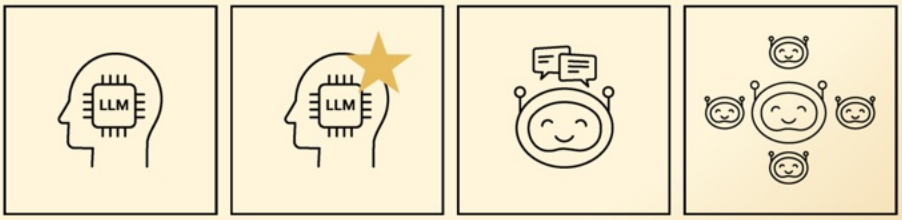
The guide offers a structured approach for moving from early experimentation toward steady and confident adoption, helping leaders develop a shared vocabulary and a thoughtful plan for readiness.

What is an AI agent?

DEFINITION

The minimum viable definition of an agent is a large language model equipped with at least one tool. The tool can be anything from an API to an MCP to a SaaS connector. AI agents are like digital employees, with roles, access, data and the ability to make decisions in pursuit of goals in real time.

ChatGPT vs. CustomGPT vs Agent vs. Multi-System



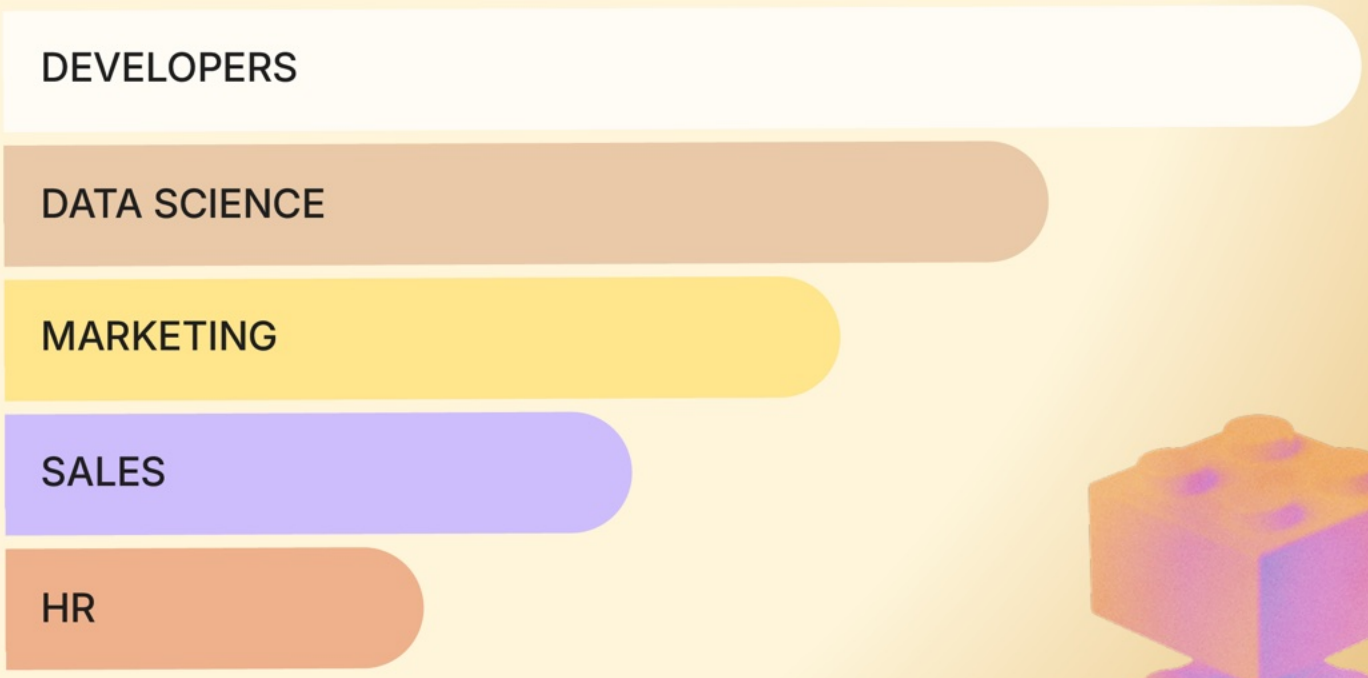
> We stop debating intent and look at capability and autonomy. The moment an LLM can observe state, decide a next action, and execute that action—especially across system boundaries—it's an agent in our threat model. It doesn't matter whether it's branded a 'copilot' or 'workflow helper.' If it can take actions without a human approving each step, we treat it as an agent and subject it to agent-level controls.
>
> Leo Cunningham, CISO @ Owkin

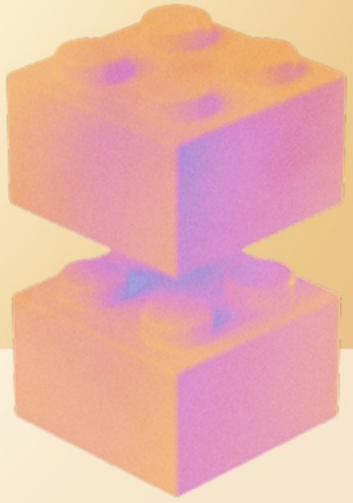
10 Good Examples of Agents

that you might not realize are already live in your environment

1. ChatGPT web or desktop + Calendar or Shared Drive connectors
2. Claude (or Claude Code) + HubSpot, GitHub, or Notion connectors
3. GitHub Copilot + the GitHub Actions API
4. Cursor + GitHub connector
5. Copilot Studio Agent connected to ServiceNow or Jira Service Management API
6. Claude + Salesforce or HubSpot connectors
7. AWS Bedrock + Confluence or Bitbucket connectors
8. OpenAI Codex + Azure DevOps connector
9. Custom GPT + Confluence, Jira, or ServiceNow connectors
10. Google Gemini + internal ITSM APIs

Top 5 departments we're seeing using agents





What's the difference between adopting ChatGPT or Custom GPT and agents?

Adopting ChatGPT, creating custom GPTs within the OpenAI platform, or any other LLM service (like Google Gemini, Anthropic's Claude, or Microsoft Copilot) creates a chat interface where, based on prompts, the services can:

- Summarise information
- Create documents or images
- Retrieve data in response to queries

The difference between these LLM services and agents is that:

- Agents can actually accomplish tasks and take actions (rather than simply responding to queries or generating content)
- Agents, especially when they're less autonomous, can still be prompted, but the range of actions they can take extends far beyond the prompt
- They can make decisions based on their instructions, taking prompts into account, which diverge greatly from the prompt's intention, including chaining tools and collaborating with other agents

Predetermined Logic vs. Decision-Making

One of the key attributes of automation and traditional software is that it operates based on logic: predetermined decisions that follow a series of if-then statements and conditions.

Conversely, autonomous AI, like AI agents, makes decisions in real-time based on the circumstances and context they encounter. Agents don't operate based on decision trees (if-then statements) and instead are non-deterministic, meaning their decisions are unpredictable since they happen in real-time (much like humans).

Agents pursue goals and make decisions in pursuit of those goals. One example that demonstrates the difference between AI agents and traditional software is to compare how each handles phishing triage in a SOC. A traditional SOAR playbook uses predetermined logic. It extracts indicators from an email, checks them against threat-intel sources, and follows a fixed sequence of if/then rules. Given the same inputs, it always produces the same outcome. This makes it predictable and easy to audit, but also limits its ability to handle novel or ambiguous attacks that fall outside the scripted workflow.

An AI agent takes a very different approach. With an LLM and tool access, it reads the full email, interprets intent, reasons about context, and decides which actions to take next, such as querying external sources, reviewing past incidents, or drafting an internal update. Its behavior adapts to what it observes rather than following a single predefined path. This increases effectiveness in unfamiliar scenarios, but introduces new risks including inconsistent decisions, unclear reasoning, and the potential for unapproved actions or data access. Traditional software executes rules. AI agents make decisions.

> Once you understand that agents operate differently from traditional automation, you stop treating them like sophisticated workflows. Playbooks can't secure systems that choose their next action based on context. Instead, you focus on which decisions you're delegating, how tightly they're bounded, and how you'll monitor behavior over time.
>
> Antonio Bovoso, Veteran Cybersecurity Leader

The Capability Continuum

Near/proto-agents

LLMs with instructions but without tools; or LLMs with the building blocks of agent structure but with no autonomy

Agents with tools

An LLM + connectors, APIs, or MCPs, have bounded autonomy, and the ability to execute tasks; can be prompted or autonomous

Orchestrated multi-agent systems

Multiple agents where agents are both executing individual tasks, being used to govern other agents, and used as resources [agent-as-tool] in complex workflows to accomplish larger capabilities and more complex functions; can be semi-to-fully autonomous

Autonomous systems

Fully autonomous without human prompting required and minimal oversight; capable of fulfilling entire functions and capabilities using numerous agents, orchestrators, agents as tools, and tools, all with dynamic and independent decision-making

Operating Model for Agentic Systems

A helpful way to think about agents is that they operate like digital employees, since they are:

- Granted systems access and permissions
- Have specific role remits and tools to complete their tasks
- Can cooperate with other agent peers to perform wide-reaching functions

They are also goal-oriented, and operate across multiple modalities of tools such as APIs, Model Context Protocol (MCP) servers, and Agent to Agent (A2A): leveraging existing application and data pathways as well as building new agent-specific protocols and methods.

Model Context Protocol (MCP)

MCP comes up frequently within AI security conversations because of both its novelty and its high usage across agentic platforms and workflows. MCP is a standardised protocol that defines and shares meaning for agents in the form of context. They enable AI agents and tools to not only exchange data, but also align on intent and context. The purpose of an MCP is to serve as a tool within an AI agent's workflow and to assist when chained together with other tools in complex autonomous workflows and systems.

MCPs are different from APIs in that they:

- Focus on understanding, not just transfer
- Enable interoperability across intelligent systems
- Support dynamic negotiation and adaptation

MCPs are complex and, like agents, require constant monitoring for drift. Their scope and definition of what is acceptable can change over time, as can how agents interact with them.

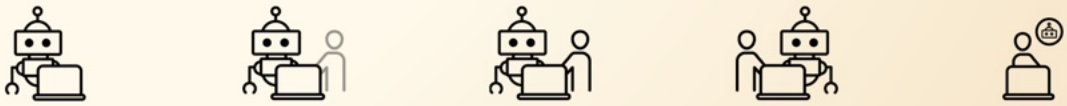
For most agentic systems, the operating model will move from individual agents, or small numbers of agents, to multi-agent systems, where agents not only operate together as teams, but are even leveraged as tools (agents as tools) for broader orchestration and large-scale tasks.

> We are experimenting with treating agents like digital employees. Each agent has an owner, a defined job, explicit permissions, and regular review of what it actually did. It is not perfect, but it's far better than trusting default settings and hoping for the best.
>
> Terry O'Daniel, Security Leader @ Netflix, Salesforce, Yahoo, Instacart, and Amplitude

Levels of Autonomy

This will also involve changes in the amount of bounded autonomy designed for and given to agents.

A good system to follow is Stanford's Human Agency Scale which helps to explain the levels of autonomy in different AI systems like AI agents, where decisions are all made in real-time by the systems, but the degree of control changes based on the autonomy built into the system. For example, in a customer service use case, a business might want the human support workers to drive the tasks, with AI augmenting the workflows such as finding the right tickets, recommending answers faster, or triaging for priorities. In a vulnerability management use case, a business may want the AI agent to identify and patch vulnerabilities without needing the human workforce to do any of the tasks, freeing them up for other work.



	HAS H1	HAS H2	HAS H3	HAS H4	HAS H5
Team Dynamics	**AI Agent Drives Task Completion** The AI agent takes primary responsibility for task execution with no or minimal human oversight.		**Equal Partnership** The human and the AI agent collaborate closely throughout the task.	**Human Drives Task Completion** The human takes primary responsibility for task execution with varying levels of AI assistance.	
Required Human Involvement	AI agent handles the task entirely on its own without your involvement.	AI agent needs your input at a few key points to achieve better task performance.	AI agent and you work together to outperform either alone.	AI agent needs your input to successfully complete the task.	Task completion fully relies on your involvement.
AI Role	**Automation** AI replaces human capabilities		**Augmentation** AI enhances human capabilities		
Example Tasks	• Transcribe data to worksheets and enter data into computer. • Run monthly network reports.	• Devise trading, option, or hedge strategies. • Accept payment on accounts.	• Create core game features, including storylines, role-play mechanics, etc. • Compile and analyze experimental data and adjust experimental designs as necessary.	• Coordinate and direct the financial planning, budgeting, procurement, or investment activities. • Design, plan, organize, or direct orientation and training programs.	• Participate in online forums or conferences to stay abreast of online retailing trends, techniques, or security threats.

***Levels of Human Agency Scale (HAS)**. Adapted from Stanford University SALT lab, Future of Work with AI Agents - Data Explorer (https://futureofwork.saltlab.stanford.edu/data-explorer)*

These new systems will build on existing policies and access points such as operating within Zero Trust principles or authentication via identity providers or non-human identity (NHI) solutions; however, their actions and decisions cannot be measured exactly in the same manner as humans.

Agentic Threat Landscape

Agentic threats mark a new chapter in enterprise security.

They share familiar foundations with traditional risks such as data exposure, access misuse, and system compromise. What has changed is the scale and complexity of the systems involved. Traditional security threats depend on external manipulation, but agentic systems can generate risk on their own. Their ability to reason, plan, and act autonomously introduces new failure modes where harm arises from misalignment rather than intent.

How Agents Amplify Existing Security Risks

Some of these threats are extensions of what security teams already know. These include:

- Identity hijacking through delegated credentials
- Prompt injection across systems, such as emails or document payloads
- Data exfiltration through unsafe tool use or memory corruption

Many of the controls remain the same. Security teams still focus on:

- Validating identity
- Managing access
- Protecting data

What has evolved is the vector of attack. Where once a threat actor tricked a system into executing malicious code, now the target is the agent's interface:

A single corrupted memory entry, poisoned API response, or tampered document can lead an agent to take unintended actions. The system does not need to be breached in a conventional sense. It only needs to make an incorrect inference based on compromised information.

> "An agent uses valid credentials, calls approved tools, and produces clean logs, yet can still end up violating our policy. Our focus shifts from 'did someone break in' to 'did the system make a decision we wouldn't have signed off on.'
>
> Ben Dewar-Powell, CISO @ AISI

New Agentic Security Risks

Unique agentic threats are specific to agents' autonomy: their decision-making can cause real-time impacts and widespread errors that, given the scale of agentic operations, can be 10x or even 100x that of a human.
These threats include: • Autonomous decision making • Tooling chaining • Cascading failures

This new and challenging class of agentic threats comes from within. Agents can fail silently, acting logically within their own parameters, while producing outcomes that violate policy or leak sensitive information. These silent failures often go unnoticed because they appear legitimate at the transactional level.

Logs may show valid requests and successful API calls, yet the combined behaviour diverges from organisational intent. In connected environments, these small deviations can compound into larger systemic failures. Each introduces risk without a single, observable act of compromise.

This dynamic gives rise to cascading agentic errors. One agent misinterprets context or a signal, leading another to act on flawed information. Each decision seems valid in isolation, yet the chain of actions produces unwanted outcomes. A reporting agent may summarise restricted financial data, which an operations agent then uses to communicate with an external partner. Every component behaves correctly according to its rules, but the overall process results in exposure and non-compliance.

Existing Security Risks Amplified: Extensions of threats already known

Traditional Risks
- Data exposure
- Access misuse
- System compromise

Traditional security controls
- Validating identity
- Managing access
- Protecting data

Amplified for agents

Extended Threats
- Identity hijacking through delegated credentials
- Prompt injection across systems, such as emails or document payloads
- Data exfiltration through unsafe tool use or memory corruption

Security gap

Evolved vectors of attack specifically targeting the agent's interface
- An agent making an incorrect inference based on compromised information (no system breach necessary)

Examples
- A single corrupt memory entry
- Poisoned API response
- Tampered document

+

New Agentic Security Risks: Unique threats to agents' autonomy

Security gap
Agents can fail silently, acting logically within their own parameters while producing outcomes that violate policy or leak sensitive information.

Examples
- Autonomous decisions
- Tools chaining
- Cascading failures

Geordie

Adoption & Transition Playbook

Most CISOs begin their agentic AI journey through executive-led initiatives, often driven by board-level interest in adopting AI across the business.

These programs typically start with ambition and experimentation, but not always with the structures in place to guide adoption responsibly.

In more mature enterprises, this early momentum has developed into formal governance efforts. Cross-functional committees are being established, inputs are being gathered across departments, and governance frameworks are beginning to take shape to ensure that AI use is consistent, transparent, and aligned with strategic goals.

Early Adoption: Safe and Simple Starts

Early adoption often begins at the periphery of the organization, where AI can quickly demonstrate value in controlled settings. These are usually low-risk, high-friction areas such as:

1. Meeting summarization
2. Note-taking
3. Internal knowledge base management

These smaller pilots help teams explore agentic AI in practical contexts while building internal understanding and confidence.

The Hidden Complexity

CISOs are often surprised by how much AI activity already exists within their organizations. Developers are using coding assistants to streamline workflows, while business users experiment with low- and no-code tools like Microsoft's Copilot Studio, which allows agents to be built directly through chat interfaces.

This organic growth reveals how rapidly AI capabilities are expanding and how easily they embed themselves into daily operations. It also highlights the growing need for visibility and alignment across functions, as new tools and systems enter the enterprise environment at an accelerating pace.

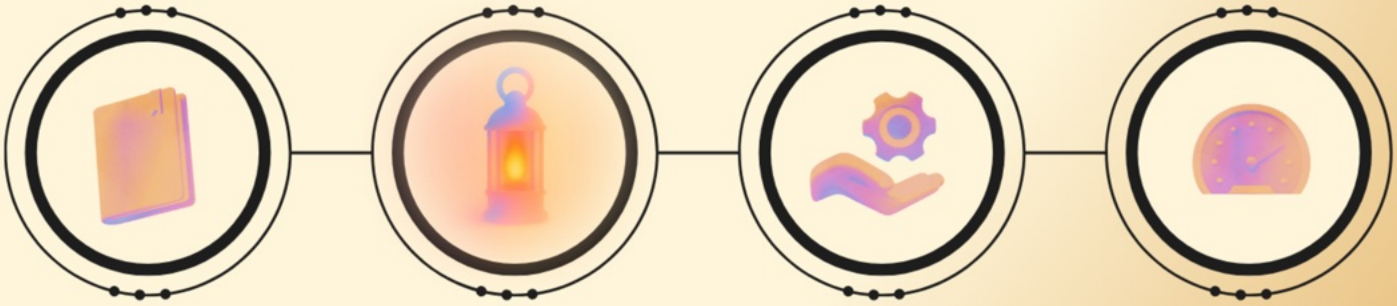
A good point of reference is the 10 Good Examples of Agents on Pg. 2

> “
>
> It grew out of experiments in engineering and ops teams that were trying to remove friction from their own work. The playbook that helped us most is to first get an honest view of where agents and AI tools exist, then understand what access they actually have, and only after that, start considering governance.
>
> Ben Dewar-Powell, CISO @ AISI

From Pilots to Enterprise Scale

Once governance foundations are established, the next phase of the journey is about operational consistency and enablement. This stage focuses on ensuring that AI systems are deployed in a coordinated and transparent way, supported by clear accountability and shared visibility across teams.

Scaling agentic AI successfully requires structures that empower experimentation while maintaining oversight. The goal is to enable innovation at every level of the business through a foundation of trust, control, and clarity.



Step 1
Establish Strategy and Frameworks

Step 2
Pilots and Experimentation

Step 3
Operationalising and Refining Governance

Step 4
Scale

The 3 Agentic Governance Priorities CISOs Must Get Right

1. **Clarity & Visibility:** Gain line of sight into the business's existing AI tool use, and where teams are using or building agents across systems and platforms.
2. **Posture & Behaviour:** Understand which tools agents are configured to use, what your agentic posture is, and what agents are doing operationally, and measure that against business expectations.
3. **Risk Insight & Governance:** Specific, context-based risk intelligence and real-time mitigations that can act at agent-scale to keep agents operating safely beyond basic guardrails.

Geordie

See how Geordie helps secure AI agents in your environment.

geordie.ai/book-a-demo

THANK YOU