CEQUENCE

Top Considerations for Enterprise Agentic AI Projects

AI has rapidly moved from hype to reality. As agentic AI projects proliferate, enterprises are discovering that many initiatives fail to deliver measurable business value. Industry research indicates that a majority of generative AI pilots are stalling or failing due to unclear ROI, governance gaps, security risks, and workflow integration challenges.

These outcomes are not unusual for emerging technologies, but they highlight a critical need: enterprises must approach agentic AI with the same rigor applied to other mission-critical systems. Based on real-world deployments and lessons learned from large enterprise customers, the following considerations outline how organizations can accelerate value from agentic AI while maintaining a strong security and governance posture.

Cequence Security has a unique perspective on how to address this challenge due to its experience protecting enterprise applications and data and how entities, both human and synthetic, interact with them. The Cequence AI Gateway enables organizations secure, govern, and control agentic workflows end to end, from the LLM to the enterprise applications agents interact with. Since its introduction, several of the largest Cequence customers have used the AI Gateway to evaluate and deploy cutting edge agentic AI solutions. Based on that experience, we have developed the following list of considerations for enterprises undertaking agentic AI projects.

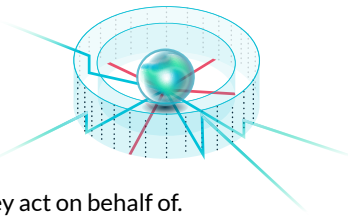
1.

Know Your Full AI Footprint First

Security teams can't govern agents, MCP servers, or LLM providers they don't know exist, and business teams are adopting all three faster than most security reviews can track. One 2,000-employee financial services enterprise ran its first inventory and found 740% more shadow MCP servers, 800% more AI agents, and 600% more LLM usage than the security team had expected. Discovery should run against existing SIEM logs rather than require a new agent, proxy, or browser extension, so an accurate picture is available immediately.

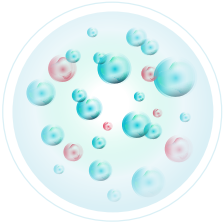
2.

Limit Agentic Access



While identity is critical, it only tells you who an entity is, not what privileges are appropriate. Agents, by default, should not have all the privileges of the user they act on behalf of. Like human workers, agents need a clear job description that frames their assigned task. That plain-language description compiles into an Agent Persona, binding the agent to a least privilege access selection of tools, APIs, LLMs, and guardrails its job requires, with everything outside that scope enforced as off-limits by policy rather than caught later in a manual review. The result is minimized risk, better performance, and lower operational costs.

3.



Govern Agentic Actions, Not Just Identity

Authentication and authorization are necessary components of secure agentic workflows, but alone are insufficient. The principles of zero trust — never trust, always verify, enforce least privilege — must now extend to agent behaviors and actions. Organizations need infrastructure that can make sure, in real time, that an agent's actions are in line with its job description. That means monitoring every action an agent takes as well as the overall context of those actions over time as behavior.

4.

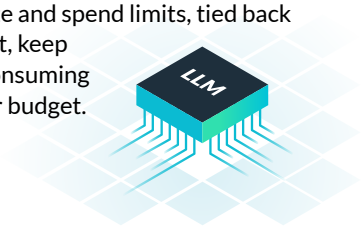
Broker API Access Instead of Distributing Credentials

Agents that call backend systems directly need a credential to do it, and every credential provided to an agent or embedded in its configuration is one that can leak or be misused. An API Registry lets agents call approved business systems without ever directly accessing the origin credential, authenticating instead through a single access key, either through one invocation tool for web-based agents or natively through the gateway's proxied endpoints. If an agent is compromised or a prompt is manipulated into misbehaving, there is no backend secret for it to expose, because the agent and the model never had one.

5.

Govern Every Call to and from the LLM

Every request an agent sends to a model, and every response that comes back, is a point where data can leak or an attacker can plant instructions. An LLM Registry should broker credentials across approved providers so agents authenticate through the gateway instead of holding a real provider API key, then apply inline data loss prevention against jailbreak attempts, system-prompt extraction, and evasion tactics such as base64-encoded payloads or invisible Unicode characters. Token-level usage visibility and enforceable rate and spend limits, tied back to the persona driving each request, keep a single misbehaving agent from consuming an unbounded amount of tokens or budget.



6.

Protect Sensitive Data

Agentic access to enterprise applications threatens to expose sensitive internal data. Organizations must be able to monitor, redact, and block sensitive data flowing through MCP tool calls, both inbound requests and outbound responses. Ideally, this capability integrates with existing DLP infrastructure and is implemented in a way that is scalable, reliable, and doesn't introduce latency.

7.

Secure and Govern MCP Server Usage

The emergence of the Model Context Protocol (MCP) has simplified agent integration with tools and enterprise data, but it also introduces new risks. Direct access to vendor-provided or public MCP servers reduces enterprise visibility and control, while malicious or poisoned MCP servers create additional attack vectors. Organizations need a trusted MCP Registry: a catalog of vetted servers built from official application APIs as well as custom MCPs for internal systems, so every server an agent reaches has already been checked against corporate security standards and data governance requirements before the first call is made.

8.

Curate a Governed Skill Registry

Without a shared catalog, every team writes its own agent instructions, and the same capability gets vetted and re-vetted every time a new use case needs it. A Skill Registry gives security and platform teams one governed set of capabilities to draw from: vetted once, reusable across every agent that needs it, and tracked so the organization can audit which skills any given agent was granted and when. That record is also what makes automatic policy mapping possible. Before a catalog like this exists, mapping a tool or API to a persona is a manual judgment call every time someone wants a new use case.

9.

Enforce Guardrails and Policy Controls

Because agents can act autonomously and at scale, guardrails are essential. Network access controls, rate limiting, and policy enforcement help ensure agents remain within defined operational boundaries. Risk-scoring mechanisms can identify anomalous or dangerous behavior and automatically throttle or halt activity.

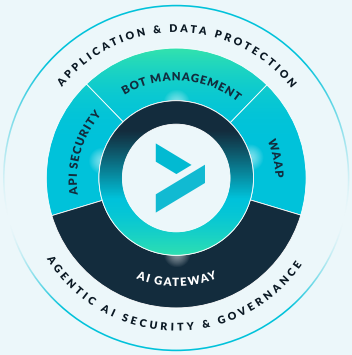
10.

Continuous Monitoring and Visibility

As enterprises deploy agentic AI, attackers are increasingly using similar technologies to automate abuse, exfiltrate sensitive data, manipulate prompts, and exploit APIs. Development teams cannot address these risks alone. Security teams must have real-time visibility into agent behavior, usage patterns, and data access in order to support audits, detect anomalies, prevent misuse, and reduce unintended expansion of the attack surface.

The Cequence Advantage

The Cequence AI Gateway makes applications agent-ready while securing and controlling agentic AI workflows, enabling organizations to unlock AI-driven productivity and growth. Built-in governance and guardrails constrain agent behavior using capabilities that include least privilege access, rate-limiting, and sensitive data protection. AI Gateway enables organizations to swiftly innovate, going from prototype to production without incurring the technical debt and scalability limitations associated with basic solutions.
Visit **www.cequence.ai/products/ai-gateway** to learn more.



CEQUENCE

5201 Great America Pkwy, Suite 240, Santa Clara, CA 95054 | 1-650-437-6338 | info@cequence.ai | www.cequence.ai