Agentic AI Secure Dev Checklist

A practitioner worksheet for AI agents, RAG apps, MCP-connected tools, automations, and workflows that can touch real systems or sensitive data.

How to use: Mark Done when satisfied, N/A when it does not apply, and leave blank when follow-up is needed.

01	Scope, ownership, and risk classification	Done	N/A
1	The agent has a named business owner, technical owner, and security reviewer.	☐	☐
2	The use case is documented, including what the agent is allowed and not allowed to do.	☐	☐
3	The agent is classified by risk level: informational, internal workflow, sensitive-data access, or production action-taking.	☐	☐
4	High-impact actions are identified, including deletes, privilege changes, payments, external messages, customer-impacting changes, and production modifications.	☐	☐

Briefly explain any N/A checks

02	Identity, access, and privilege boundaries	Done	N/A
1	The agent uses a dedicated identity, not a shared human account.	☐	☐
2	Permissions are least-privilege and scoped to the exact systems, records, repositories, or APIs required.	☐	☐
3	Credentials are short-lived where possible and stored in an approved secrets manager.	☐	☐
4	Privileged actions require step-up approval, human approval, or a separate controlled workflow.	☐	☐
5	Access is reviewed on a recurring schedule and removed when the agent is retired.	☐	☐

Briefly explain any N/A checks

03	Data, RAG, and sensitive content protection	Done	N/A
1	Approved data sources are listed, owner-approved, and separated from untrusted public content.	☐	☐
2	Sensitive data exposure is reviewed before content is indexed, embedded, retrieved, summarized, or sent to a model.	☐	☐
3	Retrieval results are permission-filtered so users cannot use the agent to access content they should not see.	☐	☐
4	Prompts, outputs, logs, and traces are reviewed for accidental storage of secrets, PII, credentials, regulated data, or confidential material.	☐	☐
5	Data retention, deletion, and export behavior is documented.	☐	☐

Briefly explain any N/A checks

04	Prompt injection and untrusted input defenses	Done	N/A
1	The agent treats webpages, documents, emails, tickets, chats, and tool outputs as untrusted input.	☐	☐
2	System instructions clearly tell the agent to ignore instructions found inside retrieved content or external data.	☐	☐
3	Tool calls are constrained by allowlists, parameter validation, and server-side authorization checks.	☐	☐
4	The agent cannot reveal hidden prompts, credentials, system instructions, private context, or other users' data.	☐	☐
5	Prompt-injection tests are performed using malicious documents, webpages, and tool responses before release.	☐	☐

Briefly explain any N/A checks

05	Tools, APIs, MCP servers, and automation safety	Done	N/A
1	Every connected tool has an owner, purpose, approved actions, and maximum impact documented.	☐	☐
2	Tool execution is logged with user, agent, timestamp, action, input parameters, and result.	☐	☐
3	Dangerous tools are separated from read-only tools and require stronger approval gates.	☐	☐
4	MCP servers and plugin-style connectors are treated as privileged integration points and reviewed before use.	☐	☐
5	The agent is blocked from chaining actions into unapproved workflows or escalating privileges through tool combinations.	☐	☐

Briefly explain any N/A checks

06	Human approval, UX warnings, and operating limits	Done	N/A
1	Users can clearly tell when they are interacting with AI-generated output.	☐	☐
2	The agent explains uncertainty and does not present guesses as verified facts.	☐	☐
3	Human approval is required before irreversible, external, regulated, financial, legal, security-sensitive, or production-impacting actions.	☐	☐
4	Rate limits, spending limits, tool-use limits, and runaway-task protections are configured.	☐	☐
5	The agent has a kill switch or disable path that support/security teams know how to use.	☐	☐

Briefly explain any N/A checks

07	Testing, red teaming, and release gates	Done	N/A
1	The agent has been tested against misuse cases, unexpected inputs, jailbreak attempts, and prompt injection.	☐	☐
2	Security tests include unauthorized data access, unsafe tool execution, excessive autonomy, and identity abuse.	☐	☐
3	Regression tests exist for critical prompts, retrieval behavior, tool permissions, and safety controls.	☐	☐
4	Production release requires approval from the product/business owner, technical owner, and security reviewer.	☐	☐
5	A rollback plan exists if the agent behaves unexpectedly after release.	☐	☐

Briefly explain any N/A checks

08	Monitoring, auditability, and incident response	Done	N/A
1	Logs capture prompts, tool calls, key decisions, approvals, errors, and blocked actions at an appropriate sensitivity level.	☐	☐
2	Alerts exist for unusual access, repeated tool failures, high-risk actions, policy bypass attempts, and anomalous volume.	☐	☐
3	Incident response playbooks include AI-specific scenarios such as prompt injection, data leakage, rogue automation, and connector compromise.	☐	☐
4	The team can reconstruct what the agent saw, decided, and did during a security review.	☐	☐
5	Model, prompt, connector, and data-source changes are change-controlled.	☐	☐

Briefly explain any N/A checks

Practitioner sign-off

System / Agent name	Owner	Reviewer	Date