RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

BACK TO BASICS

SECURING AN AGENTIC AI SYSTEM

This document is intended for organisations that deploy, configure, or integrate an agentic system, including off-the-shelf products whose configuration and connectors it controls. It is aimed at architects, CISOs, operators, and developers with a sound understanding of agentic system architectures.

Securing an agentic AI system relies on **isolation**, **access control**, and **deterministic controls** which cannot be overridden by agents. **An authorised action may still cause harm**: an agent may make mistakes, be manipulated, or dangerously combine permitted operations. Deterministic controls enforce the safeguards defined by design, and constrain nondeterministic model decisions that could exceed them.

Seven principles bring together the organisational and technical measures needed to secure an agentic system.

1/ ANALYSING AND TREATING RISKS END TO END

→ **Conduct a risk assessment from the design stage**, covering the agentic system, its effects, the chosen model, and the runtime environment. Update the assessment after each change.

→ **Define appropriate security measures for each identified risk**, drawing on authorisation, containment, segmentation, and detection. Appoint a **designated person** to be responsible for each measure. Block the deployment of the system if there remain any unacceptable risks.

→ **Include security requirements from the risk assessment in the specifications** for every product or service forming part of the agentic system.

2/ LIMITING THE AUTONOMY OF THE AGENTIC SYSTEM

→ **Define and document the agentic system's mission during the design stage,** then at least annually or whenever a significant addition or modification is made. Regularly assess the agentic system's suitability and results against its mission and risk assessment.

→ **Inventory the agentic system's accesses, resources and capabilities**, and justify that each of them is strictly necessary to meet operational needs. Use configuration controls outside of the model to ensure that the system conforms to this inventory.

→ **Implement approved accesses and capabilities gradually.** Expand autonomy only **after verifying the effectiveness of security measures** and incident management procedures.

3/ CLASSIFYING AND CONTROLING AGENT ACTIONS

→ **Define criticality criteria, document the business impact of a malfunction, and classify possible actions** according to their purpose into one of the following three categories: autonomous execution, mandatory human approval, and prohibited action.

→ **Require human approval for critical actions** (particularly those of high-risk AI systems, as defined in Article 6 of the EU AI Act), access to sensitive data, identity or permission changes, and production deployments. Keep the approval mechanism out of the agentic system's reach.

→ **Present the exact action, its goal, and its expected effects to the human approver. Execute only the approved action.**

www.cyber.gouv.fr / conseil.technique@ssi.gouv.fr

4/ SEPARATING AND ISOLATING ENVIRONMENTS

→ **Dedicate identities, secrets, and connectors** to each environment: development, evaluation, and production. Dedicate physical resources to each environment, if justified by the risk assessment.

→ **Define separate memory spaces for each customer organisation and environment.** Distinguish individual from shared memories by identifying the users or agents with access to them.

→ **Isolate the agent's runtime, tools, and connectors** in a dedicated, isolated, hardened and regularly reviewed environment to restrict access to host data.

→ **Route external communications exclusively through controlled interfaces**, in addition to this isolation. Authenticate, encrypt and filter those communications by destination, protocol, and data type.

→ **Treat any data sent to a hosted model as potentially accessible to its provider.**

5/ BASING AUTHORISATION ON A DETERMINISTIC CONTROL

→ **Implement, where feasible, a deterministic filtering mechanism that is independent of the agent,** upstream of each tool or resource, and cannot be bypassed or modified by the agent.

→ **Authenticate the requester and automatically verify their permissions**, the requested action, its parameters and target, and compliance with business rules under a policy which cannot be modified by model instructions. **Deny the action by default.**

→ **Verify that a sequence of individually authorised actions cannot produce an outcome prohibited** by business or security rules.

→ **Do not implicitly trust agents or grant them privileges by default.**

6/ LIMITING THE IMPACTING OF AN AGENTIC SYSTEM

→ **Treat the agentic system's inputs and outputs as untrusted:** a message or memory may divert the agent from its mission. Filter inputs and outputs, set retention rules and provide mechanisms to delete, forget, or quarantine them.

→ **Limit consecutive calls between agents,** and alert on unusual call rates.

→ **Keep secrets out of the model context, memories, and logs.** Delegate their management to a vault or proxy, and grant the agent only short-lived and least-privilege tokens scoped to operational needs.

→ **Provide an emergency stop** that revokes access and can be triggered by a monitoring mechanism when processing is abnormal. Predefine a degraded operating mode without weakening controls or expanding permissions.

→ **Test recovery mechanisms to return to a controlled state.** Ensure the system does not replay, after recovery, an action that has already produced an external effect.

7/ PROUVING AND MAINTAINING CONTROL OF THE SYSTEM

→ **Maintain an inventory of the agentic system's software components, system prompts, and dependencies**. Version internally developed components. Regularly use this inventory to verify each component's provenance and integrity, and check for known vulnerabilities affecting it.

→ **Test the system before deployment and regularly thereafter** to validate its security and detect model drift. Ensure that a rollback mechanism exists.

→ **Monitor agent actions in production** by comparing them against the documented mission and the authorised tools in order to detect shifts in the agent's objectives and unusual action sequences (UEBA - User and Entity Behavior Analytics).

→ **Protect the agentic system logs from modification by the agent** to ensure their integrity and enable reconstruction of the full sequence of agent activity in the event of an incident:
request → authorisation → action → result.

→ **Maintain an up-to-date register of those responsible for the agentic system's policies, alerts, and exceptions.** Update it after any change or incident and at least annually, taking the system's criticality into account.