

Agentforce Incident Response Guidelines

Before handling any Agentforce incidents, first review the [Federal Agency's Incident Response Team \(IRT\): Incident Management Process](#) so you know how incidents are classified, who to notify, what approvals are required, and how your actions fit into the Federal Agency's overall response framework.

Preparedness

- Maintain a scope and dependencies map, an up-to-date diagram of Agentforce agents, prompts, permissions, data paths, external APIs, and tenant boundaries.
- Set up threshold alerts for unusual agent behavior before incidents occur.
- Define normal behavior for each org so alerts can catch drift early.
- Create procedures for how to respond when agents behave unexpectedly, such as misclassifying information, hallucinating, drifting, or acting unpredictably.
- Classify incidents into AI-specific categories, including hallucination, data leakage, prompt injection, unauthorized action, model degradation, compliance violation, or availability outage.
- Define severity levels, decide who gets notified or takes action, and document processes for each incident.

Detection and Monitoring

- Log agent interactions, prompts, invocable actions, and outcomes, including agent, prompt template, and model versions, and configuration metadata.
- Monitor for hallucinations, harmful responses, data exposure, excessive API/invocable action usage, prompt injection attempts, and workflow failures.
- Trigger alerts for version drift, Flows and external API failure spikes, kill-switch usage, agent fallback rates, validation failures, recovery time, guardrails, blocked actions, and request volume spikes.

Containment

- Use Agentforce "kill-switches" to disable an agent or specific subagent topics or actions, or remove high-risk actions from the agent's allowed action set.
- Use Salesforce platform "kill-switches" to stop system-level behavior by disabling Flows or invocable actions, freezing the integration user, revoking OAuth tokens, restricting permissions, or blocking external API-driven workflows.
- Use layered kill-switches for external systems/APIs, network-level systems, identity and access management, data systems, and messaging systems to provide immediate containment when incidents extend beyond the Salesforce environment.
- Enforce least-privilege access by separating high-risk invocable actions from conversational capabilities.
- Configure agents using least-privilege access principles, limiting them to the minimum actions and data required to reduce security and compliance risk.
- Require human approval for financial transactions and sensitive record changes or workflows.

Investigation and Analysis

- Preserve the full record of trace data, including user prompts, agent responses, invocable action logs, data source access, and Flows and kill-switches triggered.
- Determine root cause by analyzing prompt design, agent configuration, model behavior, integration failures, data quality issues, user misuse, or security attacks.

Remediation

- Validate improved prompts and policy changes in sandbox environments, retest the reconfigured agent, refine guardrails and grounding data, and regression test all affected scenarios before redeployment.

Recovery

- Restore agents in stages by beginning with read-only access, then limited actions, and finally full functionality with monitoring.
- Track incident metrics like recovery time, kill-switch use, how often the agent falls back to a human, and validation failures during the monitoring period.

Documentation and Auditability

- Maintain incident logs, investigation reports, root-cause analyses, corrective actions, and detailed inventories of agent capabilities, invocable actions, Flows, integrations, and kill-switch processes.
- Ensure audit logs capture kill-switch events, containment steps, and recovery decisions.

Testing and Sustainment

- Hold regular practice runs covering agent failures and recovery steps, and routinely test kill-switches, shutdown procedures, Flow disablement, and escalation paths.
- Continuously verify that kill-switches and integration shutdowns work before enabling agents in production.