AI Agent Safety Preflight & Rollback Checklist

AI Agent Safety Preflight & Rollback Checklist

Brand: OperatorForge AI
Website: https://operatorforge-ai.pages.dev
Asset type: Free guide / lead magnet
Audience: founders, automation consultants, and operations leads who want agents to act without creating expensive cleanup work.

Why this exists

Small teams are moving from chatbots to agents that run commands, edit files, update CRMs, send messages, and trigger workflows. The recurring pain is not â can the agent do it?â It is â how do I let it do useful work without reviewing every click, command, and change forever?â

Use this checklist before giving an AI agent access to production workflows, client data, shell commands, email, CRM, finance tools, or customer-facing channels.

The 15-minute quick pass

1. What is the exact outcome? Write the desired artifact or system state in one sentence.
2. What is out of bounds? List systems the agent must not touch.
3. What can be reversed? Identify the rollback path before work starts.
4. What evidence is required? Screenshots, diffs, exported logs, test outputs, public URLs, or API reads.
5. What approval gate exists? Decide what requires human approval: spend, outbound messages, deleting records, production deploys.
6. What secret-handling rule applies? No raw API keys/tokens/passwords in chat, logs, screenshots, or generated docs.
7. What is the stop condition? Define when the agent should stop and report instead of continuing.

Access tiering model

Tier Agent can do Examples Approval needed

Tier 0 â Read-only Inspect, summarize, draft Audit docs, read public pages, inspect repo None beyond initial scope
Tier 1 â Local reversible Create/edit local files, run local checks Draft guide, generate PDF, create branch Before publishing externally

reply, pipeline change Human approval before send/update
Tier 4 â Financial/destructive Spend money, rotate secrets, delete data Paid ads, account changes, record deletion Explicit human approval every time

Default rule: keep new automations at Tier 0â 2 until the rollback path has been tested.

Preflight checklist

Scope guardrails
- [ ] Written task objective exists.
- [ ] Systems in scope are named.
- [ ] Systems out of scope are named.
- [ ] The agent knows whether it may write files, run commands, deploy, send messages, or mutate records.
- [ ] The agent knows whether it is operating under a brand, client, or internal project.
- [ ] Brand separation is clear where relevant.

Environment safety
- [ ] Work happens in a sandbox, draft workspace, branch, staging site, or reversible local directory where possible.
- [ ] Production write access is avoided unless needed.
- [ ] Commands that may hang use timeouts or background process management.
- [ ] Long-running jobs have log polling or completion checks.
- [ ] The agent is not allowed to install paid services or create accounts without approval.

Secrets and credentials
- [ ] Credentials are already available through approved local environment/config, not pasted into chat.
- [ ] Output is redacted before reporting.
- [ ] Logs do not include bearer tokens, cookies, passwords, private keys, or full API URLs with sensitive query strings.
- [ ] The agent uses only the narrow credentials needed for the named task.
- [ ] Any failed auth is reported as a blocker; the agent does not hunt through unrelated private files unless credential recovery is explicitly in scope.

Data boundaries
- [ ] Test data is clearly labeled.
- [ ] No live customer message is sent during testing.
- [ ] CRM/accounting/support mutations are made only against test records unless approved.
- [ ] Exports are stored in the intended workspace, not random temp folders.

- [ ] Sensitive records are summarized rather than copied into deliverables.

Tool permissions
- [ ] Shell commands are allowed only for the intended workspace.
- [ ] Browser actions are limited to public pages or approved logged-in systems.
- [ ] Email/social/CRM actions stop at drafts unless approval says â send/update.â
- [ ] The agent is instructed to verify live state after deploys or external changes.
- [ ] The agent must report blockers instead of bypassing permission walls.

Rollback plan template

text
Task:
Workspace/project:
Systems in scope:
Systems out of scope:
Allowed actions:
Approval-required actions:
Rollback owner:
Rollback method:
Last known good state:
Verification method:
Stop condition:

Agent run evidence checklist

- [ ] Files created or modified with paths.
- [ ] Tests/checks run and their result.
- [ ] Public URLs verified, if deployed.
- [ ] Screenshot or visual inspection for page changes when relevant.
- [ ] Known blockers and exact asks from the operator.
- [ ] Rollback instructions for any external mutation.

Red flags that should stop the run

Stop and ask for human approval if the agent needs to spend money, send real outbound messages, delete records, rotate credentials, modify DNS/payment/legal/account settings, use credentials found in unrelated files, or work around logins/paywalls/CAPTCHAs/rate limits/ToS restrictions.

Client handoff email template

Subject: Agent automation safety checklist before go-live

Hi {{clientname}},

Before we let the automation/agent act on live data, I want to run one final safety pass. The goal is to confirm what the agent can change, what stays read-only, how we roll back, and what proof we need after each run.

Iâ ll verify allowed systems, approval gates, test records, credential handling, rollback steps, and live verification evidence. If anything touches customer messages, billing, production data, or destructive actions, Iâ ll keep it in draft/test mode until you approve the exact step.

Best,
{{yourname}}

Monetization note for consultants

Package this as an Agent Safety Audit ($250â $750), Automation Go-Live QA ($500â $1,500), or Managed Agent Ops retainer ($300â $2,000/month).

Related resources:
- Web version: https://operatorforge-ai.pages.dev/guides/ai-agent-safety-preflight-rollback-checklist.html
- n8n QA Checklist: https://operatorforge-ai.pages.dev/guides/n8n-client-automation-qa-checklist.html
- OperatorForge AI launch kit: https://operatorforge-ai.pages.dev/