Aona

Claude Code fake-secret access review

Engineering security and Claude Code administrators

Read the complete guide on aona.ai

Keep real credentials outside the material a coding task needs, configure the relevant Claude Code file and tool permissions, and verify the effective boundary with fake markers. Test direct reads, search and shell access separately in the intended client and mode. A deny rule's presence is useful configuration evidence, but it is not a recorded outcome for every operation.

Review direct file, search and shell operations against the same harmless marker.

Synthetic canaries and blank observations. No current Claude Code or Aona behaviour has been tested here.

Your review

Reviewer: _____ Date: _____

Organisation or team: _____

☐ **Use a new folder with fake values only**

Do not point the exercise at a repository containing credentials or customer records.

☐ **Identify the effective rules and sandbox**

Record their source, supported platform and mode before the first operation.

☐ **Separate each operation and result**

Keep rule presence, approval decision and content returned in different columns.

Checked items record your review, not a compliance approval or an installed-product test result.

Notes and next actions

Decision and scope: _____

Evidence to collect: _____

Owner and next review date: _____

Review matrix

Fixture	Purpose	State
allowed-marker.txt	Confirm the baseline can be read	Untested
env-canary.txt	Copy to .env.canary in the test folder	Synthetic only
operations.csv	Separate permission action and output	Blank observations

Claude Code fake-secret review

Synthetic fixture only. No file applies settings, invokes Claude Code or makes API calls. No real secrets are included.

1. Use a NEW isolated folder containing only this pack. Copy env-canary.txt to .env.canary. Do not use a real repository or symlink to other data.
2. Inspect claude-settings.fragment.json. Its only rule is permissions.deny: Read(./.env.canary), documented relative to the current directory. It is not an active settings filename. An authorised administrator may merge this reviewed fragment into the isolated project's supported settings, preserving every existing or managed restriction. Do not overwrite global settings.
3. Open that fixture folder in the approved client. In /permissions, verify the exact effective Read(./.env.canary) rule and its settings source. Record client/extension version, OS, shell and mode. If the rule is unsupported or not effective, stop and record that finding.
4. Establish the permitted baseline with the built-in file-reading tool on allowed-marker.txt.
5. Request the direct built-in read of .env.canary. With the exact rule active, the intended result is denial. Record the actual decision and whether the synthetic marker appeared; no result is supplied here.
6. Separately, if permitted by the approved test, review a named-file search and a shell read of ONLY .env.canary from this folder. Read rules have documented limits for indirect subprocess operations. Record those paths separately; do not select bypass mode or relax a sandbox.
7. Use operations.csv. Distinguish the intended policy from the observed tool decision and output.

Source review: 2026-09-21. Follow current permission syntax and supported client versions. The reviewed Bash sandbox supports macOS, Linux and WSL2; do not assume native-Windows OS-sandbox protection from this fragment. This is a direct-read rule, not a universal filesystem barrier.

Operations

These records are arranged vertically for reading and printing. The Excel workbook retains the full column layout.

Record 1

Field	Record value
Date	
Client version	
Extension version	
OS	
Shell	
Permission mode	
Sandbox state	
Rule source	
Operation	allowed baseline
Expected policy	DEFINE

Field	Record value
Permission action	UNTESTED
Marker returned	UNTESTED
Evidence	

Record 2

Field	Record value
Date	
Client version	
Extension version	
OS	
Shell	
Permission mode	
Sandbox state	
Rule source	
Operation	direct file read
Expected policy	DEFINE
Permission action	UNTESTED
Marker returned	UNTESTED
Evidence	

Record 3

Field	Record value
Date	
Client version	
Extension version	
OS	
Shell	

Field	Record value
Permission mode	
Sandbox state	
Rule source	
Operation	scoped search
Expected policy	DEFINE
Permission action	UNTESTED
Marker returned	UNTESTED
Evidence	

Record 4

Field	Record value
Date	
Client version	
Extension version	
OS	
Shell	
Permission mode	
Sandbox state	
Rule source	
Operation	shell read
Expected policy	DEFINE
Permission action	UNTESTED
Marker returned	UNTESTED
Evidence	

Access review decision

Restricted fixture path: .env.canary
Data: a fake marker, not a credential

Intended restriction: _____
Effective rule source: _____
Supported sandbox and prerequisites: _____
Observed configuration: _____
Unresolved path and owner: _____
Decision: NOT YET REVIEWED
Recheck after client, extension, shell, OS or policy change.

Practice files

The complete download pack includes these original working files. Keep their original formats when running or inspecting the examples.

- Allowed marker (allowed-marker.txt)

SYNTHETIC_D05_ALLOWED_BASELINE

- Env canary (env-canary.txt)

EXAMPLE_ONLY=NOT_A_CREDENTIAL_D05_CANARY

- Claude settings.fragment (claude-settings.fragment.json)

Sources and scope

Sources checked 2026-09-21. Review the current requirements and your actual agreement before using this workbook for a real decision. Aona does not replace Claude Code filesystem permissions, and this guide does not establish coverage of every shell result or CLI operation.

Claude Code: Configure permissions. Documents rule order, effective settings and tool-specific matching.

Claude Code: Sandboxed Bash tool. Documents supported sandbox platforms, prerequisites and filesystem/network scope.

Claude Code: version-specific recursive-read report. A dated, unverified practitioner report; supports testing operations separately, not a universal current flaw claim.

Aona: AI security coverage. Requires supported installed-client paths and separates native inspection from broader control claims.