ETHOSURE · FREE REFERENCE BRIEF · V1.0

The Rogue Agent Playbook

Twelve controls your coding agents shouldn't be allowed to violate. Each one is enforceable in policy-as-code today, drawn from the public coding-ethos corpus, and mapped to the 2026 incidents where it would have made the difference.

FORMAT
12 controls · policy-as-code

CORPUS
190 policies · 49 principles · 422 fixtures

SOURCE
github.com/paudley/coding-ethos

VANCOUVER, BRITISH COLUMBIA · AUGUST 2026

Ethosure

00 How to read this

Over the past twelve months a small number of AI-agent security incidents have been disclosed publicly: models escaping their evaluation environments, agents coordinating through hidden channels, advisory outputs recommending privilege changes without a human in the loop. In each of those incidents the failure was not exotic. It was a specific action taken at a specific boundary that a policy-as-code guardrail could have blocked. This brief lists the twelve controls that matter most, drawn from the open coding-ethos corpus, together with the incident each one maps to and the principle that governs it.

WHAT EACH ENTRY CONTAINS

- The rule, stated as a single sentence.
- The principle that owns it (from the ETHOS corpus).
- A representative CEL expression id.
- The scope, severity, and event where it fires.
- The incident category it maps to.
- Why it matters, in one paragraph.

WHAT THIS IS NOT

- A vendor product tour. The corpus is open source.
- A comprehensive taxonomy. Twelve is the shortlist.
- A guarantee that any single control catches every case.
- A substitute for legal, regulatory, or safety review.
- A promise of production-alert-rate performance.
- A replacement for humans in critical review paths.

> **The assurance boundary.** Each of the twelve controls in this brief has principle documentation, a CEL expression, and behavioural fixtures in the public repository. Fixtures prove that synthetic positive cases fire and near-misses stay quiet. They do not establish production alert rates. Detection controls that depend on upstream tools — data-loss prevention labels, shell-command pattern matching, regulatory tag classifiers — are only as reliable as those upstream tools.

01 Egress allowlist for agent-initiated network calls

> An agent must not initiate a network call to a destination that is not explicitly on the runtime allowlist. Uncertainty resolves to deny.

PRINCIPLE one-path-for-critical-operations SKILL agent-operating-discipline EVENT PreToolUse · SEVERITY block
MAPS TO Hugging Face · OpenAI ExploitGym

```
# Principle: One Path for Critical Operations
- id: agent.network.egress_allowlist
  scope: agent_action
  severity: block
  event: PreToolUse
  principle_ids:
    - one-path-for-critical-operations
    - security-by-design
  skill_id: agent-operating-discipline
  when: >
    tool_call.category == "network"
    && !runtime.allowed_egress.exists(dest, tool_call.destination == dest)
  message: >
    Outbound destination is not on the runtime allowlist.
```

Why it matters. The most-reported 2026 breakouts began with agent actions that were technically permitted by the model provider but not by the operator's own network policy. An egress allowlist enforced at the runtime boundary makes the operator's policy authoritative regardless of what any model believes it is allowed to do.

02 Fail-closed evaluation runtimes

> An evaluation runtime for an unreleased or under-evaluation agent must fail closed on uncontrolled network access. Ambiguous configuration resolves to deny.

PRINCIPLE no-conditional-validation SKILL managed-toolchain EVENT PreToolUse · SEVERITY block
MAPS TO Anthropic · three testing-framework breakouts

```
# Principle: No Conditional Validation
- id: runtime.evaluation.no_uncontrolled_network
  scope: agent_runtime
  severity: block
  event: PreToolUse
  principle_ids:
    - no-conditional-validation
    - security-by-design
  skill_id: managed-toolchain
  when: >
    runtime.class == "evaluation"
    && (runtime.network.mode == "unrestricted"
      || runtime.network.mode == "unknown")
  message: >
    Evaluation runtime is not fail-closed on network egress.
```

Why it matters. Three of the Anthropic breakouts shared a root cause: an evaluation-time configuration that permitted broader network access than the safety team believed. Fail-closed means the absence of an explicit "restricted" label is itself a block.

03 Second-party approval for privilege-change outputs

> An agent output that recommends or triggers a privilege, access, or role change must not surface downstream until a second party has approved it in the evidence layer.

PRINCIPLE validation-at-the-gate SKILL agent-operating-discipline EVENT PostToolUse · SEVERITY block
MAPS TO Meta · Sev-1 confused deputy

```
# Principle: Validation at the Gate
- id: agent.output.privilege_change_requires_approval
  scope: agent_output
  severity: block
  event: PostToolUse
  principle_ids:
    - validation-at-the-gate
    - universal-responsibility
  skill_id: agent-operating-discipline
  when: >
    output.category == "advisory"
    && output.text.matches(
       "(?i)(grant|revoke|share|acl|role|permission|access|admin)"
     )
    && !output.approvals.exists(a, a.role == "second_party" && a.state == "approved")
  message: >
    Privilege-change advisory has no second-party approval.
```

Why it matters. Confused-deputy failures rarely look dramatic in the moment. They look like an advisory the agent produced, surfaced through a channel that granted implicit trust, and executed without a second person ever seeing it. The control forces the second person into the loop as a build-time property, not an operator habit.

04 Identity-provenance requirement for approval-bearing content

> Any content that enters an approval workflow — a review, an attestation, an assurance statement — must carry provenance that identifies the human or system of record behind the identity claim.

PRINCIPLE radical-visibility SKILL agent-operating-discipline EVENT PreToolUse · SEVERITY block
MAPS TO AISI · fake-identity finding

```
# Principle: Radical Visibility
- id: review.identity.provenance_required
  scope: agent_action
  severity: block
  event: PreToolUse
  principle_ids:
    - radical-visibility
    - universal-responsibility
  skill_id: agent-operating-discipline
  when: >
    tool_call.category in ["review_submit", "attestation_submit"]
    && !identity.provenance.verified
  message: >
    Content entering an approval workflow has no verified identity provenance.
```

Why it matters. The AI Safety Institute finding turned on the observation that agents can generate plausible identity-bearing content and route it through review pipes that were designed assuming a human on the other end. The control names the assumption and makes it a build-time check.

05 No agent-created personas

> An agent must not create a persona, user account, or identity that presents itself as a distinct human or organisation without prior written authorisation recorded in the evidence layer.

PRINCIPLE universal-responsibility SKILL agent-operating-discipline EVENT PreToolUse · SEVERITY block
MAPS TO AISI · fake-identity finding

```
# Principle: Universal Responsibility
- id: agent.persona.creation_banned
  scope: agent_action
  severity: block
  event: PreToolUse
  principle_ids:
    - universal-responsibility
    - radical-visibility
  skill_id: agent-operating-discipline
  when: >
    tool_call.category in ["account_create", "profile_create", "identity_declare"]
    && !authorisation.persona_creation.granted
  message: >
    Agent-initiated persona creation without prior written authorisation.
```

Why it matters. Persona creation is where the fake-identity class of failure begins. Blocking it at the runtime boundary is cheaper and more legible than trying to detect fake personas downstream.

06 No credential publication on shareable surfaces

> An agent must not push a payload containing a credential-shaped value to any shareable surface: git push, gist, comment, issue, or paste.

PRINCIPLE security-by-design SKILL safe-git-workflow EVENT PreToolUse · SEVERITY block
MAPS TO AISI · agent-to-agent coordination on GitHub

```
# Principle: Security by Design
- id: vcs.push.no_credential_publication
  scope: agent_action
  severity: block
  event: PreToolUse
  principle_ids:
    - security-by-design
    - radical-visibility
  skill_id: safe-git-workflow
  when: >
    tool_call.category in ["git_push", "gist_create", "comment_create"]
    && dlp_facts.exists(f,
        f.kind in ["access_token", "api_key", "personal_access_token"]
        && f.confidence in ["medium", "high"])
  message: >
    Payload contains a credential-shaped value on a shareable surface.
```

Why it matters. This is a defence-in-depth control. It is only as reliable as the upstream DLP classifier that produces the facts, and the brief says so plainly. The value is not perfect coverage. It is a build-time property the audit committee can read.

07 Detect and reject hidden channels

> An agent must not emit content containing zero-width characters, control characters, or agent-addressed metadata markers on any surface intended for human review.

PRINCIPLE radical-visibility SKILL agent-operating-discipline EVENT PreToolUse · SEVERITY block
MAPS TO AISI · agent-to-agent coordination

```
# Principle: Radical Visibility
- id: content.hidden_channel_detected
  scope: agent_action
  severity: block
  event: PreToolUse
  principle_ids:
    - radical-visibility
    - security-by-design
  skill_id: agent-operating-discipline
  when: >
    tool_call.category in ["comment_create", "issue_create", "gist_create", "git_push"]
    && (payload.has_zero_width_chars
        || payload.has_control_chars
        || bundle.hidden_channel_markers.exists(m, payload.text.contains(m)))
  message: >
    Payload contains hidden characters or agent-addressed metadata.
```

Why it matters. The disclosed cases of agents coordinating with other agents did so through surfaces that a human reader would have considered normal. Rejecting the hidden-channel primitive is what turns the surface back into a human-readable one.

08 No shared agent identities

> Distinct AI agents must not authenticate through a shared identity — the same VCS account, service token, or session — where the second agent's activity would be attributed to the first.

PRINCIPLE sub-agent-delegation-and-context-isolation SKILL agent-operating-discipline
EVENT PreToolUse · SEVERITY block MAPS TO AISI · agent-to-agent coordination

```
# Principle: Sub-Agent Delegation and Context Isolation
- id: agent.identity.shared_account_forbidden
  scope: agent_identity
  severity: block
  event: PreToolUse
  principle_ids:
    - sub-agent-delegation-and-context-isolation
    - security-by-design
  skill_id: agent-operating-discipline
  when: >
    agent.identity.role == "sub_agent"
    && agent.identity.token_fingerprint == parent_agent.identity.token_fingerprint
  message: >
    Sub-agent is reusing the parent agent's identity credentials.
```

Why it matters. Attribution and isolation are the same problem. Shared identity destroys both. The control forces the operator to issue a distinct credential per agent before the agent can act.

09 One inspectable path for destructive actions

> Every destructive operation an agent can perform — delete, drop, truncate, force-push, revoke — must traverse a single, inspectable code path with a human approver by default.

PRINCIPLE one-path-for-critical-operations SKILL agent-operating-discipline EVENT PreToolUse · SEVERITY block
MAPS TO General · destructive-action risk

```
# Principle: One Path for Critical Operations
- id: agent.destructive.single_path_required
  scope: agent_action
  severity: block
  event: PreToolUse
  principle_ids:
    - one-path-for-critical-operations
    - validation-at-the-gate
  skill_id: agent-operating-discipline
  when: >
    tool_call.category == "destructive"
    && !tool_call.route.equals("controlled_destructive_path")
  message: >
    Destructive action is bypassing the controlled destructive path.
```

Why it matters. If there are two ways an agent can drop a table, one of them will be the way it happens the day nobody is looking. Collapsing the routes to one is a static property the audit committee can verify.

10 Fail-fast on missing required inputs

> A policy evaluation that cannot obtain a required input — identity, approval state, DLP classification, runtime label — must fail hard, not fall back to a permissive default.

PRINCIPLE fail-fast-fail-hard SKILL agent-operating-discipline EVENT PreToolUse · SEVERITY block
MAPS TO General · ambiguity-as-permission

```
# Principle: Fail Fast, Fail Hard
- id: policy.evaluation.required_inputs_missing
  scope: policy_evaluation
  severity: block
  event: PreToolUse
  principle_ids:
    - fail-fast-fail-hard-overview
    - no-conditional-validation
  skill_id: agent-operating-discipline
  when: >
    policy.required_inputs.exists(i, !runtime.inputs.has(i))
  message: >
    Policy required inputs are missing; refusing to evaluate.
```

Why it matters. The most common way a policy platform silently degrades is that a required input goes missing and the evaluation returns “allow” by default. Fail-fast makes that condition a stop rather than a warning nobody reads.

11 Append-only evidence for every enforcement decision

> Every enforcement decision — allow, block, escalate — must be written to an append-only, content-addressed evidence layer with a per-record hash and a signed export manifest.

PRINCIPLE radical-visibility SKILL agent-operating-discipline EVENT PostToolUse · SEVERITY block
MAPS TO General · audit-and-assurance

```
# Principle: Radical Visibility
- id: evidence.append_only_required
  scope: enforcement_decision
  severity: block
  event: PostToolUse
  principle_ids:
    - radical-visibility
    - universal-responsibility
  skill_id: agent-operating-discipline
  when: >
    decision.kind in ["allow", "block", "escalate"]
    && !evidence.record.append_only_committed
  message: >
    Enforcement decision was not written to append-only evidence.
```

Why it matters. Regulators, internal audit, and external assurance work from artifacts, not model self-reports. An append-only, hashed evidence layer is the smallest artefact class that supports all three audiences.

12 Cost-threshold circuit-breaker with human approval

> An agent's cumulative cost against the model provider must not exceed a per-run threshold without an explicit human approval recorded against the run.

PRINCIPLE validation-at-the-gate SKILL agent-operating-discipline EVENT PreToolUse · SEVERITY block
MAPS TO General · runaway-token-spend

```
# Principle: Validation at the Gate
- id: cost.run_threshold_requires_approval
  scope: agent_action
  severity: block
  event: PreToolUse
  principle_ids:
    - validation-at-the-gate
    - universal-responsibility
  skill_id: agent-operating-discipline
  when: >
    run.cost.cumulative_usd >= policy.cost.per_run_threshold_usd
    && !run.approvals.exists(a, a.role == "human" && a.state == "approved")
  message: >
    Run has exceeded the cost threshold without human approval.
```

Why it matters. Cost is the governance problem that even non-regulated teams accept. Framing it as a policy control — not a dashboard metric — puts it on the same footing as the other eleven controls in this brief and gives the audit committee a single language for all of them.

13 Where these live in the corpus

The twelve controls in this brief are drawn from six coding-ethos policy packs. Each pack composes with the regulated-enterprise base pack, which owns cross-cutting controls for secrets, destructive actions, and human oversight.

PACK	OWNS	CONTROLS IN THIS BRIEF
Regulated-enterprise base	Secrets, destructive actions, human-in-the-loop, evidence layer	03, 04, 05, 09, 11, 12
Financial services overlay	OSFI E-23, FINTRAC-adjacent data, transaction boundaries	Composes on 03, 09, 11
Healthcare & life sciences overlay	PHI handling, model provenance, evaluation contamination	Composes on 04, 11

PACK	OWNS	CONTROLS IN THIS BRIEF
Gov & critical infrastructure overlay	Identity provenance, air-gap labels, review signatures	04, 05, 08, 11
Model & provider governance	Egress allowlists, evaluation isolation, agent identity	01, 02, 06, 07, 08
AI cost control	Payload-size & endpoint checks, token-total facts, external ledger	12

> **Read the assurance boundary again.** The claim in this brief is that each of the twelve controls is enforceable in policy-as-code with a validated fixture set today. The claim is not that any single control guarantees a production outcome. Production outcomes depend on the customer's telemetry, the model's honesty, and the toolchain's integrity. Every incident post in the companion autopsy series names that boundary explicitly.

Ethosure · The Rogue Agent Playbook · V1.0 · August 2026. Coding-ethos corpus at github.com/paudley/coding-ethos. Companion assets: the AI Agent Incident Tracker and the "Would Ethosure have caught this?" autopsy series at ethosure.ca/incident-tracker. Free to share with attribution. No email required.