فطرة

FITRAH · WORKING PAPER DRAFT · ZEESHAN MEHDI · 2 NOVEMBER 2026

Hereditary Trust for Autonomous Agent Lineages

Status: working paper draft. Repository evidence is canonical. This draft does not claim general AI safety or established novelty.

Abstract

Fitrah studies whether an autonomous agent can be created through an authorised birth process with a mandatory inherited constitutional core and bounded authority that descendants cannot silently weaken or amplify, independently of model cooperation.

Lab 0 separates mutable cognition from runtime-held constitutional state, exercisable authority and delegable authority. Stage 1 fixed the operational semantics through adversarial hand-written scenarios. Stage 1B added deep lineages, generated histories, fault injection and bounded exhaustive search. No counterexample was found for the correct runtime within the tested bounds: 100 generations, 2,000 generated histories covering 100,000 transitions, and a complete bounded search of 6,742 reachable states and 459,386 transitions.

These bounded findings were subsequently reproduced in GitHub-hosted workflows with retained evidence checked by a separate first-party verifier process, without changing the research claims. Stage 1B also exposed verifier defects and TC-26: replayed signed evidence establishes internal consistency, not truth against a consistently lying compromised runtime.

Stage 2 moves from hostile requests to hostile code inside a disposable Hyper-V Linux VM. The environment has demonstrated distinct non-root runtime and agent identities, denied protected filesystem access, disconnected networking and cgroup-v2 descendant containment. A captured prerequisite run reports P01-P09 PASS / READY. S65 shows ordinary process-group termination is insufficient while the cgroup boundary terminates the tested descendant escape patterns.

Integration work then exposed TC-27, an evidence/state atomicity defect, motivating transactional persistent-state work under TC-28 and an experimental birth-before-execution lifecycle choice under TC-29. The first guarded resource-backed run passed prerequisites but returned INVALID_RUN after authorised birth and startup failure. Cleanup was confirmed and no workload action occurred.

The present evidence supports only a bounded claim: hereditary constitutional equality and non-amplifying authority can be implemented and robustly tested within the stated model and environment assumptions. It does not establish general security, truth against a compromised runtime, clone resistance or novelty over relevant capability-security and attestation work.

1. Research question

Can an autonomous agent be born through an authorised process with a mandatory constitutional core and bounded authority that descendants cannot silently weaken or amplify, independently of whether the language model chooses to comply?

Lab 0 studies deterministic enforcement rather than whether a model voluntarily behaves safely.

2. Model

For Lab 0, runtime state contains finite agent records with identity, parent, constitutional core, mutable state, exercisable capabilities C, delegable capabilities Dg and liveness.

Accepted hereditary conditions include:

```
core(child) = core(root)
C(child) subset_of Dg(parent)
Dg(child) subset_of C(child)
```

Protected execution also requires the agent to be alive, to hold the capability and to be explicitly allowed by the inherited core.

The root is created from trusted genesis. Descendant birth is valid only when the parent is authorised to spawn and the proposed child satisfies the hereditary constraints.

3. Why this is not an LLM guardrail

The model is untrusted. Prompts, refusals and LLM judges are not treated as deterministic security controls.

The runtime holds protected state and mediates protected execution. The hereditary properties are therefore testable with no LLM present.

4. Stage 1

Stage 1 established the concrete semantics of core integrity, mandatory inheritance, bounded delegation, explicit-allow policy, authorised birth, rollback, evidence replay and bounded communication.

Its main value was semantic and engineering. Stronger evidence came later from generated histories, independent replay and deliberate implementation weakening.

5. Stage 1B

Stage 1B results for the correct runtime:

- lineages tested to 100 generations;
- 2,000 seeded generated histories;
- 100,000 generated transitions;
- 6,742 reachable states in the bounded model;
- 459,386 state/request transitions checked;
- no invariant counterexample within those tested bounds.

Fault injection demonstrated that the search and verifier could find violations in deliberately weakened runtimes.

After verifier repairs, baseline generated histories produced 59,408 executed violations and the verifier detected all 59,408.

TC-26

A runtime can falsify an event and its state digest consistently. A verifier may then see internally consistent evidence until later behaviour exposes a contradiction.

Therefore Lab 0 verification establishes consistency of retained evidence with the model, not truth against a compromised runtime.

6. Stage 2 execution boundary

The Stage 2 VM uses:

- separate non-root runtime and agent identities;
- runtime-only signing material;
- protected registry, evidence and resource paths;
- no agent external network during runs;
- one root-owned cgroup-v2 boundary per agent;
- a small fixed allowlist privileged helper that performs cgroup placement before dropping to the agent uid.

A captured disconnected run reports P01-P09 PASS / READY.

S65 creates descendants using fork, double fork, setsid and a new process group. Ordinary process-group termination leaves tested escapees. Cgroup teardown terminates the tested descendant tree and both cleanup oracles are empty.

7. Integration findings

TC-27: state/evidence atomicity

Fault injection showed an evidence append failure could leave an allowed protected state change without corresponding request evidence.

The detected in-process path was repaired and regression-tested. This finding shows that operating-system isolation alone is insufficient: protected state and evidence require explicit transaction semantics.

TC-28: versioned persistent state

An experimental design stores complete versioned generations of registry, resource ledger and signed evidence and publishes one CURRENT generation.

Prototype tests cover interruption and failure cases, but do not constitute a power-loss durability proof and the store is not yet the canonical runtime adapter.

TC-29: authorised birth before execution

For the experimental resource demonstration, authorised birth is committed before child execution. If startup fails, birth remains part of lineage history and failure/death is recorded where possible.

This remains an experimental lifecycle choice.

8. Current resource-backed result

The first guarded VM resource demonstration ran after fresh disconnected P01-P09 READY and S65 PASS.

Outcome: INVALID_RUN.

Retained evidence contains genesis, AUTHORIZED_BIRTH and STARTUP_FAILURE. Cleanup was confirmed, no descendant remained and no workload read/write/self-kill request occurred. Commit state was conservatively classified as uncertain.

A launch-readiness race is currently only a hypothesis.

9. What the evidence supports

Within the accepted Lab 0 model and stated assumptions, current evidence supports bounded claims that:

- hereditary core equality can be enforced independently of model cooperation;
- exercisable and delegable authority can be recursively attenuated;
- independent replay detects a broad class of enforcement failures;
- generated and exhaustive bounded exploration found no counterexample for the correct runtime within the tested domains;
- the tested separate-uid, no-network, cgroup-contained environment contains the tested hostile descendant behaviours;
- failures above the OS boundary remain important, particularly around evidence atomicity, persistent state and birth/launch ordering.

Current evidence does not establish general AI safety, host-compromise resistance, truth against a consistently lying runtime, clone resistance, migration safety, revocation correctness, population-level non-collusion or novelty.

10. Research significance

Capability attenuation and delegation are established ideas with substantial prior art.

The candidate contribution under investigation is the composition of authorised agent birth, mandatory hereditary constitutional state, recursive bounded authority, independently replayable lineage evidence and an execution boundary intended to remain effective under untrusted cognition and hostile descendants.

Novelty remains an open question requiring explicit related-work comparison.

10.1 Prior-art update and narrowed contribution

A current literature and standards review substantially narrows what this paper should claim.

Recent work already treats the language model as untrusted and moves authorization into independent infrastructure. *Delegation Without Trust* evaluates compromised sub-agents behind an external authorization broker. The Agent Identity Protocol defines verifiable multi-hop delegation in which downstream authority must not widen. IETF AIMS and NIST's agent-identity programme are converging on workload identity, delegated authorization, accountability and audit. SPIFFE/SPIRE, DICE/RATS and mature capability-token systems already provide important parts of identity, attenuation and attestation. Proof-Carrying Agent Actions demonstrates independent certificates for proposed tool execution.

Accordingly, Fitrah should not claim bounded delegation, external authorization, workload identity, attestation or proof-carrying checks as individual contributions.

The candidate research contribution is narrower and compositional: treating **authorised birth as a first-class security transition** that establishes a mandatory hereditary constitutional state, recursive authority ceiling, lineage evidence and eventually binding to a particular attested execution instance.

This claim remains a hypothesis. Stage 1B also exposes a direct risk to it: in the current simple policy language, constitutional strengthening largely reduces to capability attenuation. A stronger paper therefore requires an explicit demonstration that hereditary constitutional state can encode and enforce properties that an attenuated capability token alone cannot express.

The most useful comparative experiment is a three-system ablation:

1. broad bearer authority;
2. a modern attenuated delegation broker;
3. Fitrah authorised birth plus hereditary constraints.

The discriminating question is not whether Fitrah also blocks privilege amplification. It is whether Fitrah excludes invalid births, broken lineage, constitutional weakening or lifecycle states that the delegation-only system still permits.

10.2 Direction after negative ablations

The project name Fitrah is inspired by the Islamic concept of an innate created nature. The analogy is deliberately limited: this work does not equate software state with human fitrah, moral responsibility, faith, conscience or free will.

The engineering hypothesis is that machine reproduction may permit something that human heredity cannot guarantee deterministically: a constitutional substrate that is mechanically mandatory across every valid descendant even when mutable cognition diverges.

> Can every valid descendant in an autonomous machine lineage be required to inherit an irreducible constitutional substrate from its root, while learned cognition remains mutable, and can protected authority be made unavailable to descendants outside that hereditary chain?

E6 and E8 narrow this question substantially. A rich caveated broker plus an equivalent trusted registration state machine reproduced the tested Fitrah valid-state set, and a strong delegated credential chain reproduced the tested lineage-validity predicate. Therefore neither birth semantics nor portable lineage evidence alone is currently a defensible novelty claim.

The remaining candidate contribution is a systems property: making authorised birth, hereditary constitutional state, protected state, durable evidence and eventually execution-instance identity transactionally inseparable under hostile execution. An invalid process may still exist at the operating-system level; the proposed guarantee is that it cannot become a valid authority-bearing descendant in the protected trust domain.

This remains a hypothesis. The next evidence must come from hostile-execution lifecycle experiments and birth-to-instance binding, or from a future hereditary property that survives comparison with equivalently strong authorization and credential systems.

10.3 GitHub-hosted reproduction of Stage 1 and Stage 1B (19 October 2026)

[RESULT] After the bounded Stage 1 and Stage 1B findings were established, the unchanged canonical deterministic workloads were run through reviewed, manually dispatched GitHub Actions workflows on GitHub-hosted Ubuntu 24.04 with Python 3.12.15. All three runs used the committed source and recorded the execution ref, runtime metadata, hashes, separate execution and verifier outcomes, and GitHub artifact references. These are **infrastructure and reproducibility results**, not new experiments supporting stronger hereditary-trust or novelty claims.

Scope	Run and source commit	Hosted outcome	Retained evidence
Stage 1	Run 39276979467, `91740062df0eef1c68202cb9c8e74aeab02ab82b`	65 targeted tests passed; canonical Stage 1 execution completed; independent verification of all six logs passed	Canonical archive, artifact 12197717505 and separate regression diagnostics
Stage 1B, quick smoke only	Run 39282890148, `d14c6a2eaf9827afe9ac946d45ec8717ac3cdff8`	73 targeted tests passed; 399 cases verified; no verification errors	Quick archive, artifact 12202038043
Stage 1B, canonical full	Run 39284167068, `d14c6a2eaf9827afe9ac946d45ec8717ac3cdff8`	73 targeted tests passed; full experiment completed; 10,631 captured evidence cases independently replayed by the workflow verifier; zero reported verification errors	Full archive, artifact 12203401010, 137,399,588 bytes compressed; GitHub-reported archive SHA-256 `bbee5df05313abb5c5df6be25c23d19a87c917defa534998c6bfda329c948a20`

[EVIDENCE] The GitHub job logs report completion and successful artifact uploads. The contributing reviewer reports checking those logs and artifact metadata in PR #10's review comment, including full-artifact expiry at `2027-01-17T16:16:34Z`. The review did not download and audit every file in the full archive. A durable independently checked archive outside GitHub Actions is **not yet confirmed**.

[LIMITATION] The separate first-party Stage 1B verifier process replays signed logs and reconciles counts against recorded histories and runtime observations; it does not externally attest the executing runtime or repeat the complete exhaustive enumeration with a separate implementation. The test signing seed is public in the repository. TC-22 and TC-26 remain. GitHub-hosted reproduction on another execution environment is not independent third-party replication. The full Stage 1B experiment's bounded conclusions, Stage 2 status, and novelty assessment are unchanged.

The subsequent editorial audit distinguishes repository consistency checks from the hosted-evidence checks that remain blocked in that audit environment. It does not replace the contributing reviewer's reported log inspection.

11. Next experiments

1. Diagnose the failed resource-backed launch/readiness path.
2. Repeat the isolated resource run after any confirmed repair.
3. Complete canonical Stage 2 scenarios S60-S64 and S70 against the live protected resource.
4. Freeze Lab 0 evidence and limitations.
5. Introduce a real LLM as mutable cognition inside the enforced birth/runtime substrate.
6. Move subsequently to cryptographic identity, clone resistance and runtime attestation.

12. Current conclusion

Fitrah began as a hypothesis that trust properties could be hereditary across autonomous-agent reproduction rather than rediscovered by each model generation.

The work has converted that idea into a concrete transition system, independent verification, bounded robustness evidence and a real isolated hostile-code environment. It has also exposed implementation and evidence limits that were invisible in the initial abstraction.

The correct current conclusion is not that hereditary trust has been proved secure. It is that the hypothesis has survived increasingly adversarial and realistic tests so far, while the remaining difficulty has moved from abstract inheritance logic to execution, storage and lifecycle integration.

Repository evidence is canonical: https://github.com/c0ldskin/fitrah · Website: https://c0ldskin.github.io/fitrah/ · MIT licensed. This is a working draft, not a peer-reviewed publication.