MACHINA

TRINITY

The agent runtime that assumes the LLM will fail.

C++20	9	40+	v6.5
Safety Core	Defense Layers	Built-in Tools	Current Release

The Problem

Every agent framework gives an LLM a knife and hopes for the best.

`LLM hallucinates rm -rf /`

No rollback. State corrupted forever.

`System breaks at 3 AM`

No audit trail. grep through unstructured logs.

`Tool spawns runaway process`

No resource limits. No isolation.

`External LLM API goes down`

Entire agent freezes. No fallback.

These aren't edge cases. They're Tuesday.

Trinity Architecture

Three concerns. They never mix.

BODY

Execution

- Transactional Tx/Rollback
- Tool Registry + Sandbox
- Plugin hash verification
- Permission leases

DRIVER

Decision

- Heuristic selector (always)
- LLM policy (optional)
- Circuit breaker fallback
- 3-tier intent resolution

MEMORY

Record

- SHA-256 hash-chained audit
- WAL + checkpoint/recovery
- Deterministic replay
- BM25 + vector hybrid search

9 Layers of Defense

Defense-in-depth. Every layer independent. No single point of failure.

1. **Tx + Rollback**
   State integrity
2. **Hash-chained Audit**
   Tamper-evident history
3. **Allowlists**
   Command restriction
4. **seccomp-BPF**
   Kernel syscall filtering
5. **Permission Leases**
   Single-use tokens
6. **Plugin Hash Pinning**
   SHA-256 before dlopen
7. **Capability Gates**
   Bitmask permissions
8. **SSRF Defense**
   DNS rebinding prevention
9. **CRC32 WAL Framing**
   Crash integrity detection

+ bwrap isolation · Genesis source guard · nonce replay protection · HMAC signing · rate limiting · input sanitization

Genesis: Self-Evolution

Write → Compile → Verify → Load. New tools at runtime, safely.

01

WRITE

Source Guard blocks dangerous APIs & headers

02

COMPILE

Build .so/.dll record SHA-256 hash of binary

03

VERIFY

Constant-time hash comparison before dlopen

04

LOAD

Capability gate rejects plugins exceeding caps

Opt-in only (MACHINA_GENESIS_ENABLE=1). Off by default in production. Three independent safety gates.

Autonomic Engine

6-level GVU cycle. The system improves itself — monotonically.

L1 REFLECT — 5 min
Analyze recent experiences

L2 TEST — 5 min
Run self-tests, find gaps

L3 HEAL — 30 min
Auto-fix broken components

L4 HYGIENE — 30 min
Clean logs, compact memory

L5 CURIOSITY — 30 min
Explore capability gaps

L6 WEB — 30 min
Search and learn new knowledge

Regression Gate — bad changes blocked

Auto-Rollback — degradation auto-reverted

Monotonic — only improves, never degrades

Dual-Layer Design

C++ handles safety. Python handles intelligence.

C++20 Safety Core

- Transactional execution (Tx/Rollback)
- Hash-chained audit logs (SHA-256)
- seccomp-BPF syscall filtering
- Plugin system (.so/.dll + hash pinning)
- WAL + checkpoint recovery
- Concurrent Priority Queue
- HMAC auth + rate limiting
- Prometheus /metrics endpoint

Python Agent Runtime

- Telegram bot (Pulse Loop)
- 6-level Autonomic Engine
- ExpeL + Reflexion + Distillation
- Graph Memory 2.0 (multi-hop BFS)
- MCP bridge (external tools)
- 70+ tool aliases (Korean/English)
- 3-tier intent resolution
- 36 files, all ≤ 620 lines

How It Compares

Out-of-the-box capabilities.

Feature	Machina	LangChain	AutoGPT	CrewAI
Transactional execution	✓	—	—	—
Cryptographic audit trail	✓	—	—	—
Deterministic replay	✓	—	—	—
Kernel-level sandboxing	✓	—	—	—
Permission leases	✓	—	—	—
Plugin hash verification	✓	—	—	—
Circuit breaker	✓	—	—	—
Runtime self-evolution	✓	—	—	—
Prometheus /metrics	✓	—	—	—
Native C++ performance	✓	—	—	—

Test Coverage

14
C++ Unit
Test Suites

34
Python E2E
Test Cases

10
Golden
Replay Tests

39
Simulation
Scenarios

C++ Suites

CPQ · WAL · WAL Rotation · Tx · Tx Patch · Memory · Memory Query · Toolhost · GoalRegistry · Input Safety · Sandbox · Lease · Config · Plugin Hash

E2E Groups (13)

Chat Intent · Shell · Web Search · Code Exec · Memory · File Ops · Config · URL Fetch · Utilities · Chat Response · Summary · Continue Loop · Auto-Memory

MACHINA TRINITY

Built for a world where LLMs are powerful but imperfect.

Language	License	Release	Tests
C++20 + Python	Apache 2.0	v6.5	97 total

github.com/machina-trinity