

# AIR Blackbox — EU AI Act Compliance Report

## Haystack by deepset (Updated with Maintainer Feedback)

**Date:** March 13, 2026 | **Scanner:** AIR Blackbox v1.2.2 | **Author:** Jason Shotwell **Reviewed by:** Julian Risch, Haystack Maintainer (deepset-ai/haystack#10810)

### Methodology

AIR Blackbox v1.2.2 performs static pattern matching against Python source code. It detects the presence of compliance-relevant patterns mapped to EU AI Act Articles 9-15. This version includes corrections based on direct feedback from the Haystack engineering team, reducing false positives and improving detection accuracy.

**This report is a starting point for conversation, not a compliance certification.**

### Summary

**24 Passing · 10 Warnings · 5 Failing · 39 Total Checks**

95% automated detection · EU AI Act Articles 9, 10, 11, 12, 14, 15

Haystack scored highest among 6 major AI agent frameworks scanned. The Haystack team validated our findings and provided corrections that improved scanner accuracy for the entire ecosystem.

Rank	Framework	Pass	Warn	Fail	Total
#1	Haystack (deepset)	24	10	5	39
#2	Semantic Kernel (Microsoft)	15	4	0	19
#3	GPT Researcher	15	3	0	18
#4	OpenAI Agents SDK	14	5	0	19
#5	Mem0	13	6	0	19
#6	DSPy (Stanford)	12	6	1	19

## What Haystack Does Well

**Strong input validation (44%).** 245 out of 552 Python files use Pydantic models or dataclasses. Well above average.

**Highest logging coverage (26%).** 143 out of 552 files include structured logging. Best of all frameworks tested.

**Dedicated human-in-the-loop module.** 47 files with HITL patterns. Haystack has a purpose-built `human_in_the_loop` package with strategies, policies, and protocols — not bolted on.

**Robust retry/backoff (41 files).** Production-grade resilience against API failures.

**Production tracing built-in.** `HAYSTACK_CONTENT_TRACING_ENABLED` env var plus `logging_tracer.py` provides real production audit capability. Confirmed by maintainer.

**Docstrings at 29%.** The maintainer noted this is partly because some methods are public but should be private — marking them private is planned for the next major release. Effectively higher than reported.

---

## What Needs Attention

**Missing governance documents.** No `RISK_ASSESSMENT.md`, `DATA_GOVERNANCE.md`, `MODEL_CARD.md`, or `OPERATOR_GUIDE.md`. The Haystack team is discussing internally how to add these (Issue #10810, P3 label).

**Type hints at 25%.** 1,273 out of 4,996 public functions. Gradual improvement recommended.

**LLM call error handling at 23%.** 68 out of 302 files with LLM calls have try/except. Most missing coverage is in test files.

---

## OAuth Delegation Checks — Validated by Maintainer

v1.2.2 includes corrections based on Julian Risch's review. Each finding below includes the maintainer's response.

### 1. Agent-to-User Identity Binding — CONFIRMED VALID

**What we detected:** `user_id` in 3 files (telemetry, HITL strategies, memory store)

**Maintainer response:** `user_id` is used to store and retrieve memories from the memory store per user. This is real identity binding — the system tracks which user's memories are being accessed.

**Assessment:** Valid finding. Stronger than initially assessed.

### 2. Token Scope / Permission Validation — PARTIALLY VALID

**What we detected (v1.2.0):** Generic `scope` matches in 8 files **What we detect (v1.2.2):** `confirmation_strategy_context` and HITL policy patterns in 5 files

**Maintainer response:** `confirmation_strategy_context` is for passing request-scoped resources to HITL confirmation strategies in web/server environments (WebSocket connections, async queues). This is request-scoping, not OAuth permission scoping.

**Assessment:** The pattern serves a different purpose than OAuth scoping, but does demonstrate request-level isolation. Scanner updated to use more specific patterns.

### 3. Token Expiry / Execution Bounding — CONFIRMED VALID

**What we detected (v1.2.0):** `max_age` and `ttl` in 12 files (mostly cache TTL — false positives) **What we detect (v1.2.2):** `max_agent_steps` in 6 files

**Maintainer response:** Our regex matched `max_agent_steps`, which is an execution boundary that prevents endless agent actions.

**Assessment:** Valid finding. `max_agent_steps` is exactly the kind of safeguard Article 14 cares about. Scanner updated to remove cache TTL false positives and specifically detect execution boundaries.

### 4. Agent Action Audit Trail — CONFIRMED VALID (PRODUCTION)

**What we detected (v1.2.0):** `execution_log` in 2 test files **What we detect (v1.2.2):** `CONTENT_TRACING_ENABLED` and `logging_tracer` in 8 files

**Maintainer response:** Tool invocations are logged in production via `HAYSTACK_CONTENT_TRACING_ENABLED`. Built-in `logging_tracer.py` plus integration options (OpenTelemetry, Datadog, etc.) work out of the box.

**Assessment:** Much stronger than initially assessed. This is real production audit capability, not just test coverage. Scanner updated.

### 5. Agent Action Boundaries — CORRECTED

**What we detected (v1.2.0):** `is_allowed` in `serialization.py` (1 file) — **FALSE POSITIVE** **What we detect (v1.2.2):** `human_in_the_loop/policies.py` patterns in 5 files

**Maintainer response:** Confirmed `is_allowed` in `serialization.py` is about deserialization safety, not action boundaries. Pointed us to `human_in_the_loop/policies.py` as the actual action boundary

implementation.





**Assessment:** Original finding was a false positive. Scanner corrected. The real action boundaries exist in the HITL policies module.

Updated OAuth Summary






Check	v1.2.0 (before)	v1.2.2 (after feedback)	Maintainer Assessment
Identity binding	3 files (mixed)	3 files (memory store)	Confirmed valid
Scope validation	8 files (generic)	5 files (HITL strategies)	Partially valid
Execution bounding	12 files (cache TTL)	6 files (max_agent_steps)	Confirmed valid
Action audit trail	2 files (tests)	8 files (production tracing)	Confirmed — production
Action boundaries	1 file (false positive)	5 files (HITL policies)	Corrected — real boundaries

# Article-by-Article Results






## Article 9 — Risk Management

Check	Status	Evidence
Risk assessment document	 FAIL	No RISK_ASSESSMENT.md found
Risk mitigations active	 FAIL	0/4 runtime mitigations
LLM call error handling	 WARN	68/302 files (23%)
Fallback/recovery	 PASS	61 files






## Article 10 — Data Governance

Check	Status	Evidence
PII detection	 PASS	Gateway active
Data governance docs	 FAIL	No DATA_GOVERNANCE.md
Data vault	 FAIL	No vault configured
Input validation	 PASS	245/552 files (44%)
PII handling	 PASS	25 files

## Article 11 — Technical Documentation

Check	Status	Evidence
README	 PASS	Found
Runtime inventory	 PASS	Gateway observed
Model card	 WARN	Not found
Docstrings	 FAIL	29% (maintainer note: private methods inflate denominator)
Type hints	 WARN	25%

## Article 12 — Record-Keeping

Check	Status	Evidence
Event logging	 PASS	Gateway active
Audit chain	 WARN	No signing key
Log traceability	 PASS	run_id, model, timestamp
Application logging	 PASS	26% — highest of all frameworks
Tracing	 PASS	28 files

Action audit trail	✅ PASS	Production tracing via CONTENT_TRACING_ENABLED (confirmed)
--------------------	--------	--

## Article 14 — Human Oversight

Check	Status	Evidence
HITL (gateway)	⚠️ WARN	No approval gates in traffic
Kill switch	⚠️ WARN	No guardrails configured
Operator docs	⚠️ WARN	No OPERATOR_GUIDE.md
HITL (code)	✅ PASS	47 files — dedicated module
Usage limits	✅ PASS	38 files
Identity binding	✅ PASS	Memory store user_id (confirmed)
Scope validation	✅ PASS	HITL confirmation strategies (5 files)
Execution bounding	✅ PASS	max_agent_steps (confirmed)
Action boundaries	✅ PASS	HITL policies (corrected from serialization FP)

## Article 15 — Robustness

Check	Status	Evidence
Injection protection	✅ PASS	Gateway scanning
Error resilience	✅ PASS	0% error rate
API access control	⚠️ WARN	No API keys in env
Adversarial testing	⚠️ WARN	No evidence
Retry/backoff	✅ PASS	41 files
Injection defense	✅ PASS	17 files
Unsafe input	⚠️ WARN	13 files
Output validation	✅ PASS	27 files

## How This Report Was Generated

```
pip install air-blackbox==1.2.2
git clone https://github.com/deepset-ai/haystack.git
air-blackbox comply --scan ./haystack -v
```

v1.2.2 includes corrections from Haystack maintainer feedback (deepset-ai/haystack#10810).

**GitHub:** [github.com/airblackbox/gateway](https://github.com/airblackbox/gateway) **PyPI:** [pypi.org/project/air-blackbox/1.2.2](https://pypi.org/project/air-blackbox/1.2.2) **Issue:** [github.com/deepset-ai/haystack/issues/10810](https://github.com/deepset-ai/haystack/issues/10810)

Report generated by AIR Blackbox v1.2.2 — The flight recorder for AI agents. Validated with feedback from the Haystack engineering team. Contact: [jason.j.shotwell@gmail.com](mailto:jason.j.shotwell@gmail.com) | [github.com/airblackbox](https://github.com/airblackbox)