

Memory Poisoning Defense in Autonomous Agents

Graph Topology & Vector Clustering - Executive Brief

VERSION: v1.1 (August 2026)

AUTHORS: Timofej Bogdanov & SynptoSec Research

SOURCE: synptosec.com/research

DRAFT - PENDING LEGAL REVIEW. This text is not yet in force.

Summary

Mathematical formulation of slow-poisoning attacks on agent vector memories (MINJA). Explains how graph community detection identifies adversarial coordinate drift before tool execution.

In this brief

- Vector store attack surface: cross-session contamination via RAG & tool returns
- Mathematical limitations of static cosine similarity filtering
- Graph community detection algorithm for detecting stealthy malicious clusters
- Summarizes the MINJA authors' empirical evaluation (50-turn synthetic multi-agent conversations) alongside our graph-topology defense approach

From the published research pages (synptosec.com/research)

We measure our own system, test it against public benchmarks, and build on published research. Our original research program is planned.

Memory poisoning is an active research area. We implement the approach and measure our results under a documented, reproducible protocol. We do not claim to have solved the problem.

Our planned Vector SIEM feature detects MINJA-style memory poisoning through graph topology analysis. It tracks embedding clusters over time, detects anomalous cluster formation, and re-scores memory at retrieval. The approach is informed by published work in structural anomaly detection for agent memory, including MemAudit and MemSentry.

About this document

This executive brief is distilled from the published page listed above, which remains the full, current version. It contains no data beyond what is published there. These are self-published whitepapers - cite as SynptoSec whitepapers.

