

Adaptive AI Security Mesh

Technical Report

Tushal Chandwani , Rudra Pratap Singh , Sahil Aman

8Bit Odyssey

November 23, 2025

Abstract

Modern organizations increasingly integrate multiple foundation models, custom large language models, and third-party inference endpoints into mission-critical workflows. This proliferation introduces novel attack surfaces such as prompt injection, data exfiltration, model supply-chain compromise, misalignment exploitation, and policy drift. This report presents Adaptive AI Security Mesh, a unified, adaptive security layer for heterogeneous AI model environments. It orchestrates threat intelligence ingestion, autonomous red teaming, dynamic policy synthesis, real-time model probing, automated mitigation workflows, and a trust registry. Implemented as a TypeScript/Next.js service with modular security engines, WATCHTOWER MESH provides extensibility, explainability, and low-friction integration.

Contents

1	Problem Statement	3
2	Objectives	3
3	System Overview	3
4	Component Responsibilities	4
4.1	Threat Intelligence	4
4.2	Red Team AI	4
4.3	Model Probe Engine	4
4.4	Policy Generator	4
4.5	Guardian AI	4
4.6	Auto Mitigation	4
4.7	Untrusted Model Registry	5
5	API Summary	5
6	Data Flow	5
7	Security Strategy	5
7.1	Defense-in-Depth	5
7.2	Risk Scoring	5
7.3	Adversarial Simulation	6
8	Implementation Notes	6
8.1	Technology Stack	6
8.2	Code Sketches	6
9	Evaluation	7
9.1	Metrics	7
9.2	Findings	7
10	Differentiators	7
11	Limitations	7
12	Future Work	7
13	Conclusion	8

1 Problem Statement

As organizations scale multi-model deployments, traditional perimeter security is inadequate for addressing emergent risks related to language model behavior and cross-model interaction. Key challenges include:

1. Untrusted model integrations.
2. Model behavior drift.
3. Prompt-based attack vectors.
4. Static policy maintenance burden.
5. Fragmented telemetry.
6. Slow detection-to-remediation loops.

2 Objectives

- Establish a continuously adaptive AI security mesh.
- Automate governance policy lifecycle based on threat intelligence.
- Integrate autonomous red teaming.
- Provide unified APIs for probing, registration, and remediation.
- Enable minimal-latency policy enforcement and rollback.
- Maintain developer velocity with low coupling.

3 System Overview

Adaptive AI Security Mesh follows a layered architecture:

- a) Core Mesh Orchestrator.
- b) Threat Intelligence Engine.
- c) Red Team AI.
- d) Model Probe Engine.
- e) Policy Generator.
- f) Guardian AI.
- g) Auto Mitigation.
- h) Security State Store.
- i) Untrusted Model Registry.
- j) Groq Service Adapter.

4 Component Responsibilities

4.1 Threat Intelligence

Ingests vulnerability disclosures, jailbreak patterns, and malicious prompt signatures. Normalizes indicators:

$$I = \{(v_i, c_i, s_i, t_i)\}$$

where v_i is the vector classification, c_i confidence, s_i severity, and t_i temporal decay.

4.2 Red Team AI

Generates adversarial scenarios using mutation operators:

$$\text{Scenario} = f_{\text{template}}(\text{goal}, \text{constraints}, \text{obfuscations})$$

4.3 Model Probe Engine

Executes test suites and captures deviations:

$$R_m = \{(q_j, r_j, \sigma_j, \Delta_j)\}$$

4.4 Policy Generator

Transforms telemetry into structured policy statements:

$$P = g(R, I) \rightarrow \text{Ruleset}$$

4.5 Guardian AI

Real-time enforcement:

$$D = h(P, C_{\text{context}}, A_{\text{risk}})$$

4.6 Auto Mitigation

Applies automated responses including:

- Endpoint quarantine.
- Rule tightening.
- Session invalidation.
- Capability downgrading.

4.7 Untrusted Model Registry

Tracks provenance and trust score:

$$T_m = \alpha Q_m + \beta H_m + \gamma E_m - \delta I_m$$

5 API Summary

- /api/health
- /api/dashboard
- /api/model-probe
- /api/models/list
- /api/models/register
- /api/security-test
- /api/threat-intel
- /api/stream/security

6 Data Flow

1. Threat intelligence updates the knowledge base.
2. Red Team AI produces new probes.
3. Probe Engine executes tests and records anomalies.
4. Policy Generator synthesizes new rules.
5. Guardian AI enforces policies.
6. Auto Mitigation responds to high-severity events.
7. Registry adjusts model trust scores.

7 Security Strategy

7.1 Defense-in-Depth

Layered validation includes input normalization, semantic scoring, policy gating, heuristics, and escalation.

7.2 Risk Scoring

Combines static metadata and dynamic observations with temporal decay.

7.3 Adversarial Simulation

Includes synonym substitution, encoded payloads, and multi-turn exploitation.

8 Implementation Notes

8.1 Technology Stack

- Next.js (App Router)
- TypeScript
- Modular engine classes
- CSS-based dashboard UI

8.2 Code Sketches

Listing 1: Policy synthesis

```
1 class PolicyGenerator {
2   generate(results, intel) {
3     const clusters = this.cluster(results);
4     return {
5       version: Date.now(),
6       rules: clusters.map(c => ({
7         id: c.signature,
8         match: c.pattern,
9         action: "block",
10        rationale: c.explanation
11      }))
12    };
13  }
14 }
```

Listing 2: Guardian decisioning

```
1 class GuardianAI {
2   decide(ctx, policy, riskScore) {
3     const matched = policy.rules.filter(r => r.match.test(ctx.content));
4     if (matched.length) return { action: "block", reason: matched[0].rationale };
5     if (riskScore > ctx.thresholds.escalate)
6       return { action: "escalate", reason: "Risk threshold exceeded" };
7     return { action: "allow" };
8   }
9 }
```

9 Evaluation

9.1 Metrics

- Detection latency
- False positive rate
- Policy convergence
- Probe coverage
- Mitigation response time

9.2 Findings

Adaptive rule generation reduces manual intervention and improves coverage with multi-stage adversarial probes.

10 Differentiators

- a. Adaptive governance
- b. Unified mesh architecture
- c. Continuous adversarial feedback
- d. Integrated trust registry
- e. Composable modular engines

11 Limitations

- Dependent on threat feed quality
- High probe cost under peak load
- Risk of over-tightened policies

12 Future Work

- Reinforcement learning for probe generation
- Cryptographic model attestation
- Multi-modal prompt security
- Federated incident pattern sharing
- Differential privacy for telemetry

13 Conclusion

Adaptive AI Security Mesh transitions AI security from reactive scanning to a proactive adaptive mesh. Through continuous threat intelligence integration, adversarial probing, policy synthesis, and automated mitigation, the system enhances overall AI governance and defense.