Practical AI Red Teaming

Latent & Behavioral Frameworks

A practitioner-focused guide for security teams to understand, execute, and interpret advanced red teaming against large language models — without requiring ML Ops expertise.

Audience: Security Engineers, Red Teamers, Detection Engineers, AI Security Architects
Scope: Open-weight models, black-box APIs, agentic and multi-agent systems

How to Read This Guide

This document is designed to be read non-linearly. Use it as a reference while running tests, reviewing outputs, or explaining findings to leadership.

Each framework answers a different question:

Framework	Primary Question	Used When
v1: Latent Space	Where is the model structurally steerable?	You have internal access
v2: Behavioral	How does the model fail in practice?	Black-box or production testing

Core Concepts (With Intuition)

Entropy

Entropy measures uncertainty. In LLM red teaming, entropy tells you whether the model is confident, confused, or unstable. Low entropy often means the model is locked into a behavior; high entropy can mean decision-boundary grazing or instability.

> **Red Team Insight:** Low entropy can be just as dangerous as high entropy — it may indicate overconfident but steerable behavior.

MLP vs Attention

MLPs reshape representations; attention routes information. Most modern exploits work by manipulating routing, not amplification.

> **Common Mistake:** Assuming safety failures require numerical instability inside MLPs.

One Transformer Block: What Actually Happens

- Tokens are converted into embeddings (vector representations).
- Attention routes information between tokens based on relevance.
- Residual connections accumulate changes rather than replace them.
- MLP blocks reshape the representation space.
- The updated hidden state flows to the next block.

Key Insight: Safety behavior is not stored in one place — it emerges from repeated routing and accumulation.

Framework v1: Latent Space Red Teaming

Framework v1 examines whether internal mathematical structures could allow controlled steering or collapse.

Metric	What It Means	How to Interpret
Condition Number (κ)	Numerical sensitivity	High $\kappa \neq$ exploitability
σ_min	Collapse direction	Near zero indicates rank loss
σ_max	Amplification	Large values are rare in hardened models
CKA	Layer similarity	High similarity may indicate redundancy

Framework v2: Behavioral Red Teaming

Framework v2 identifies real-world failure modes using only text interaction.

Technique	What It Finds	Operational Risk
Decode Fragility	Knife-edge prompts	Sampling instability
Multi-turn Drift	Context accumulation	Agentic exploitation
Attention Sinks	Routing hijacks	Prefix injection
KV Persistence	Long-lived context	Delayed activation attacks

Using Both Frameworks Together

Behavioral failures often occur without internal instability. Latent analysis explains why some behaviors are repeatable, while others are noise.

Rule of Thumb: If v2 finds a bypass, use v1 to determine whether it is structural or incidental.

Worked Examples (To Be Expanded)

This section will include real red-team findings, showing how behavioral observations map to latent properties and inform defensive decisions.

Next: Insert real test outputs here as case studies.