Visual Demonstration Script

I. Overview and Graph State (00:00 - 00:17)

The visualization displays our scaled **Neurosymbolic Graph** of 35 nodes. The visible parameters at the top confirm the **Entropy Threshold (0.2)** and the **Adaptive Activation Threshold (0.5012)**. The red nodes are flagged because their risk exceeds one of these two thresholds, demonstrating our **Dynamic Visual Gating**.

II. Service 1: Node Details & XAI (00:18 - 01:06)

We use **Node Details & XAI** to inspect critical components. The output immediately provides the **Semantic Tripla Explanation** for each node:

- **Node 11 (NORMAL):** No alarm threshold exceeded.
- **Node 13 (RED):** Flagged due to **Statistical Risk** (entropy= 0.4995). The entropy is significantly above the threshold.
- **Node 17 (NORMAL):** No alarm threshold exceeded.

Crucially, the bottom section shows the **MSU Local Partition**. This confirms the system only loads local neighbors (e.g., 8/35 nodes loaded for Node 11), proving the **computational efficiency** and isolation capability of the MSU mechanism.

III. Service 2: Advanced NL Query (01:08 - 02:06)

The **NL Query** module simulates the **Transformer-based NLP**.

- We use a complex **AND/OR** query to filter the graph's attributes.
- The output provides the **Literal SPARQL Query** that would be sent to the backend, demonstrating the translation protocol.
- The final list confirms the nodes that satisfy the complex semantic criteria.

IV. Service 3: Causal Threat Tracing (02:07 - 02:43)

We execute **Causal Threat Tracing** to quantify the path risk. We are tracing the path between two flagged nodes: the source (Node_13) and the propagated target (Node_17).

- **Path Identification:** The system finds the most probable path (by maximizing the weight product): **Node 13 -> Node 11 -> Node 17**.
- **P-Threat Result:** The Propagation Probability is calculated: 0.5777.
- **XAI Breakdown:** The segmented explanation details the contribution of each edge.

V. Service 4: What-If Analysis (02:44 - 03:17)

FInally,we use **What-If Analysis** to test a defense strategy. We identified the edge Node_13 -> Node_11 (Weight is 0.74):

- **Action:** We simulate patching this link, reducing its weight from 0.74 to a safer **0.3**.
- **Recalculation:** The system recalculates the risk on the same path (Node_13 -> Node_17).
- **Result:** The new probability drops significantly to **0.2348**.

This demonstrates how this UI provides **data-driven decision support**: the simulated mitigation confirms that patching that single vulnerability reduces the overall path risk by over 50% (from 0.5777 to 0.2348), justifying the deployment of the security action.