**Slow Pod Kill with Dependency Injection (Service Dependency Simulation)**Simulates network latency in the frontend service by incrementally increasing the delay (from 10% to 80% correlation).

Kills a frontend pod randomly to observe system behaviour.

Partitions network traffic between frontend and ledger writer, causing them to be unable to communicate.  
  
**Why do we need this?**

Chaos engineering helps identify weaknesses in your system under real-world failure conditions. Specifically:

* **Network Latency Simulation:** Helps check how the frontend behaves when there is a delay in network responses.
* **Pod Kill Experiment:** Ensures that if a frontend pod crashes, the system can recover automatically (via Kubernetes auto-scaling, retries, etc.).
* **Network Partition Test:** Tests if the ledger service can function correctly when frontend connectivity is lost.  
    
  **Path:** /root/Kalyanicat Slow Pod Kill with Dependency Injection **Solution for These Issues**
* **Latency Tuning:** Implement retries, circuit breakers (Istio/Linkerd), and caching.
* **Pod Resilience:** Ensure auto-scaling and health checks are in place.
* **Network Partition Handling:** Implement failover mechanisms or degraded mode operation to keep the system functional.