

# Circuit Breaker & Resilience - Discussion Topics

---

## Architecture & Design

---

### 1. When should you use circuit breaker vs retry vs both?

- Transient vs persistent failures
- Combining patterns effectively
- Order of pattern application

### 2. How do you determine the right failure threshold for a circuit breaker?

- Balancing sensitivity vs stability
- Service-specific considerations
- Monitoring and tuning approaches

### 3. What are the trade-offs between thread pool isolation and semaphore-based bulkheads?

- Resource overhead
- Timeout handling
- Use cases for each approach

## Real-World Scenarios

---

### 1. Design resilience patterns for a payment processing system

- Which patterns are critical?
- Fallback strategies for financial transactions
- Idempotency considerations

### 2. How would you handle a cascading failure in a microservices architecture?

- Detection and alerting
- Automatic vs manual intervention

- Recovery strategies

### **3. What happens when your circuit breaker opens during peak traffic?**

- Capacity planning implications
- Fallback capacity requirements
- User experience considerations

## **Implementation Challenges**

---

### **1. How do you test resilience patterns effectively?**

- Chaos engineering approaches
- Simulating failures in staging
- Load testing with failure injection

### **2. What metrics should you monitor for circuit breakers?**

- State transitions
- Failure rates and patterns
- Latency distributions

### **3. How do you handle circuit breaker state in a distributed system?**

- Local vs shared state
- Consistency across instances
- Coordination challenges

## **Advanced Topics**

---

### **1. How do you implement graceful degradation without losing critical functionality?**

- Feature prioritization
- Partial response strategies
- User communication during degradation

### **2. What's the relationship between circuit breakers and rate limiting?**

- Complementary patterns

- When to use each
- Combined implementation strategies

### **3. How do you handle retry storms in a distributed system?**

- Jitter strategies
- Backpressure mechanisms
- Coordinated retry policies