Name = Aditya singh
Roll NO 2301010229

Assignment 4

1. Distributed Deadlock Detection simulation

→ Fragments :-
- S_1 : $P_1 \to P_2$ $P_3 \to P_4$
- S_2 : $P_2 \to P_5$, $P_5 \to P_6$
- S_3 : $P_6 \to P_1$

(a) Global wait for graph (combined

$$P_1 \to P_2 \to P_5 \to P_6 \to P_1 \text{ (cycle)}$$

Also, $P_3 \to P_4$ (separate)

(b) Deadlock ?

→ Yes, processes involved in deadlock cycle ; P_1, P_2, P_5, P_6

c suggested distrubuted algorithms

→ use the chandy → mishra → Hass edge -chasing (probe) Algorithm for disrrubted deadlock detection each sits sends probes along wait for edges to detect cycles without centralized graph assombly.

2 Distributed file system performances:

Given: local = 5ms, remote = 25 ms
prob(remote) = 0.3

(a) Expected across time:-

$$E = 0.7 \times 5\text{ms} + 0.3 \times 25\text{ms}$$
$$= 3.5 + 7.5$$
$$= 11.\text{ms}$$

(b) coaching strategy :-
client side read cache with LRU + TTL based validatation

→ Justification:- Frequently-read remote files will be served locally redducing remote access (0.3 fraction), LRU evicts less used items. TTL keeps stateness bounded. Improves average extency while keeping consistency manageable

3 check pointing mix to meet RPO = 1s

→ Given)

full = 200 ms, incremental = 50 ms, RPO = 1s

(a) proposed mix (over 10s) :-

- Take one full checkpoint every 10 s (at t = 0 in period)
- Take incremental checkpoints every 1s (at t = 1.2---9

Total overhead (per 10s) : 1×200 ms + 9×50 ms = 200 + 450 ⇒ 680 ms

(b) Reasing :-

- with increamental 1s, the maximum worl lost on failure ≤ 1s ⇒ meets RPO.
- full once per 10s bound recovery line
- This mix minimizes full checkpoint cost wh keeping incremented frequency high enough to meet RPO

9 case study - Global e-commerce platform:-

(a) Distributed scheduling challenges Iro flash sales:-

- massive, sudden split in request
- Geographic distribution & latency - data locality matters for latency and inventory correcthess
- Heterogeneous nodes
- Stateful server

-> suggest algorithms for lead balancing
-> Hybrid Approach

(b) Fault - tolerance Strategy (RTO & RPO):-

- Active :- active multip - region deployment : service run con currently in multiple regions so failurous is seamless (RTO ≈ neal - zero at service level)

- Data strategy

-> critical transactional data := use synchrono replication within the region to guarantee consistency and law RPO, cross region replication can be asyn chronous but with frequentt replication to keep RPO small.

-> catalog / less critical data :- use multiple region eventual consistency with frequent asyn replication and cache

- operational measures :- chaos testing, backups runbook, and automata scaling to miximise RTO, use snapstick + increamental backup to bond RPO

21/11/25