Kubernetes crash recovery commands that works 99% of the time:

1. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗴𝗲𝘁 𝗽𝗼𝗱𝘀 --𝗮𝗹𝗹-𝗻𝗮𝗺𝗲𝘀𝗽𝗮𝗰𝗲𝘀: Check the status of all pods across namespaces to identify failures.

2. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗱𝗲𝘀𝗰𝗿𝗶𝗯𝗲 𝗽𝗼𝗱 𝗽𝗼𝗱\_𝗻𝗮𝗺𝗲: Gather detailed information about a failed pod.

3. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗹𝗼𝗴𝘀 𝗽𝗼𝗱\_𝗻𝗮𝗺𝗲 -𝗰 𝗰𝗼𝗻𝘁𝗮𝗶𝗻𝗲𝗿\_𝗻𝗮𝗺𝗲: View logs of a specific container inside a pod to troubleshoot issues.

4. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗴𝗲𝘁 𝗲𝘃𝗲𝗻𝘁𝘀 --𝗮𝗹𝗹-𝗻𝗮𝗺𝗲𝘀𝗽𝗮𝗰𝗲𝘀 --𝘀𝗼𝗿𝘁-𝗯𝘆='.𝗺𝗲𝘁𝗮𝗱𝗮𝘁𝗮.𝗰𝗿𝗲𝗮𝘁𝗶𝗼𝗻𝗧𝗶𝗺𝗲𝘀𝘁𝗮𝗺𝗽': Review recent events for clues on crashes and errors.

5. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗴𝗲𝘁 𝗻𝗼𝗱𝗲𝘀: Verify the status of nodes in the cluster, checking for node failures.

‌6. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗱𝗿𝗮𝗶𝗻 𝗻𝗼𝗱𝗲\_𝗻𝗮𝗺𝗲 --𝗶𝗴𝗻𝗼𝗿𝗲-𝗱𝗮𝗲𝗺𝗼𝗻𝘀𝗲𝘁𝘀: Safely evacuate and cordon a node for recovery operations.

7. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗰𝗼𝗿𝗱𝗼𝗻 𝗻𝗼𝗱𝗲\_𝗻𝗮𝗺𝗲: Mark a node as unschedulable to prevent new pods from being scheduled during recovery.

8. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗱𝗲𝗹𝗲𝘁𝗲 𝗽𝗼𝗱 𝗽𝗼𝗱\_𝗻𝗮𝗺𝗲 --𝗴𝗿𝗮𝗰𝗲-𝗽𝗲𝗿𝗶𝗼𝗱=0 --𝗳𝗼𝗿𝗰𝗲: Forcefully delete a crashed pod to restart it or clear it for recovery.

9. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗿𝗼𝗹𝗹𝗼𝘂𝘁 𝘂𝗻𝗱𝗼 𝗱𝗲𝗽𝗹𝗼𝘆𝗺𝗲𝗻𝘁 𝗱𝗲𝗽𝗹𝗼𝘆𝗺𝗲𝗻𝘁\_𝗻𝗮𝗺𝗲: Roll back a deployment in case a new rollout causes crashes.

10. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗲𝘅𝗲𝗰 -𝗶𝘁 𝗽𝗼𝗱\_𝗻𝗮𝗺𝗲 -- /𝗯𝗶𝗻/𝘀𝗵: Access a container to debug and resolve application issues directly inside the pod.

11. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗴𝗲𝘁 𝗰𝗼𝗺𝗽𝗼𝗻𝗲𝗻𝘁𝘀𝘁𝗮𝘁𝘂𝘀𝗲𝘀: Check the health of core cluster components like etcd, kube-apiserver, and more.

‌12. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝘁𝗼𝗽 𝗻𝗼𝗱𝗲𝘀: Monitor node resource usage to detect resource exhaustion causing crashes.

13. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝘁𝗼𝗽 𝗽𝗼𝗱𝘀 --𝗮𝗹𝗹-𝗻𝗮𝗺𝗲𝘀𝗽𝗮𝗰𝗲𝘀: Check pod resource usage across namespaces, identifying bottlenecks leading to crashes.

14. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗱𝗲𝗹𝗲𝘁𝗲 𝗻𝗼𝗱𝗲 𝗻𝗼𝗱𝗲\_𝗻𝗮𝗺𝗲: Remove a failed node from the cluster to allow recovery operations.

15. 𝗲𝘁𝗰𝗱𝗰𝘁𝗹 --𝗲𝗻𝗱𝗽𝗼𝗶𝗻𝘁𝘀=𝗵𝘁𝘁𝗽𝘀://𝗲𝘁𝗰𝗱-𝘀𝗲𝗿𝘃𝗲𝗿:2379 𝘀𝗻𝗮𝗽𝘀𝗵𝗼𝘁 𝗿𝗲𝘀𝘁𝗼𝗿𝗲 𝗯𝗮𝗰𝗸𝘂𝗽.𝗱𝗯: Restore etcd from a snapshot in case of etcd failure..

‌𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗮𝗽𝗽𝗹𝘆 -𝗳 𝗯𝗮𝗰𝗸𝘂𝗽.𝘆𝗮𝗺𝗹: Reapply configurations from a backup manifest during recovery.

17. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝘁𝗮𝗶𝗻𝘁 𝗻𝗼𝗱𝗲𝘀 𝗻𝗼𝗱𝗲\_𝗻𝗮𝗺𝗲 𝗸𝗲𝘆=𝘃𝗮𝗹𝘂𝗲:𝗡𝗼𝗦𝗰𝗵𝗲𝗱𝘂𝗹𝗲: Prevent scheduling on a node experiencing issues during recovery.

18. 𝗸𝘂𝗯𝗲𝗰𝘁𝗹 𝗴𝗲𝘁 𝗲𝗻𝗱𝗽𝗼𝗶𝗻𝘁𝘀 𝘀𝗲𝗿𝘃𝗶𝗰𝗲\_𝗻𝗮𝗺𝗲: Verify service endpoints during recovery to ensure services are resolving correctly.