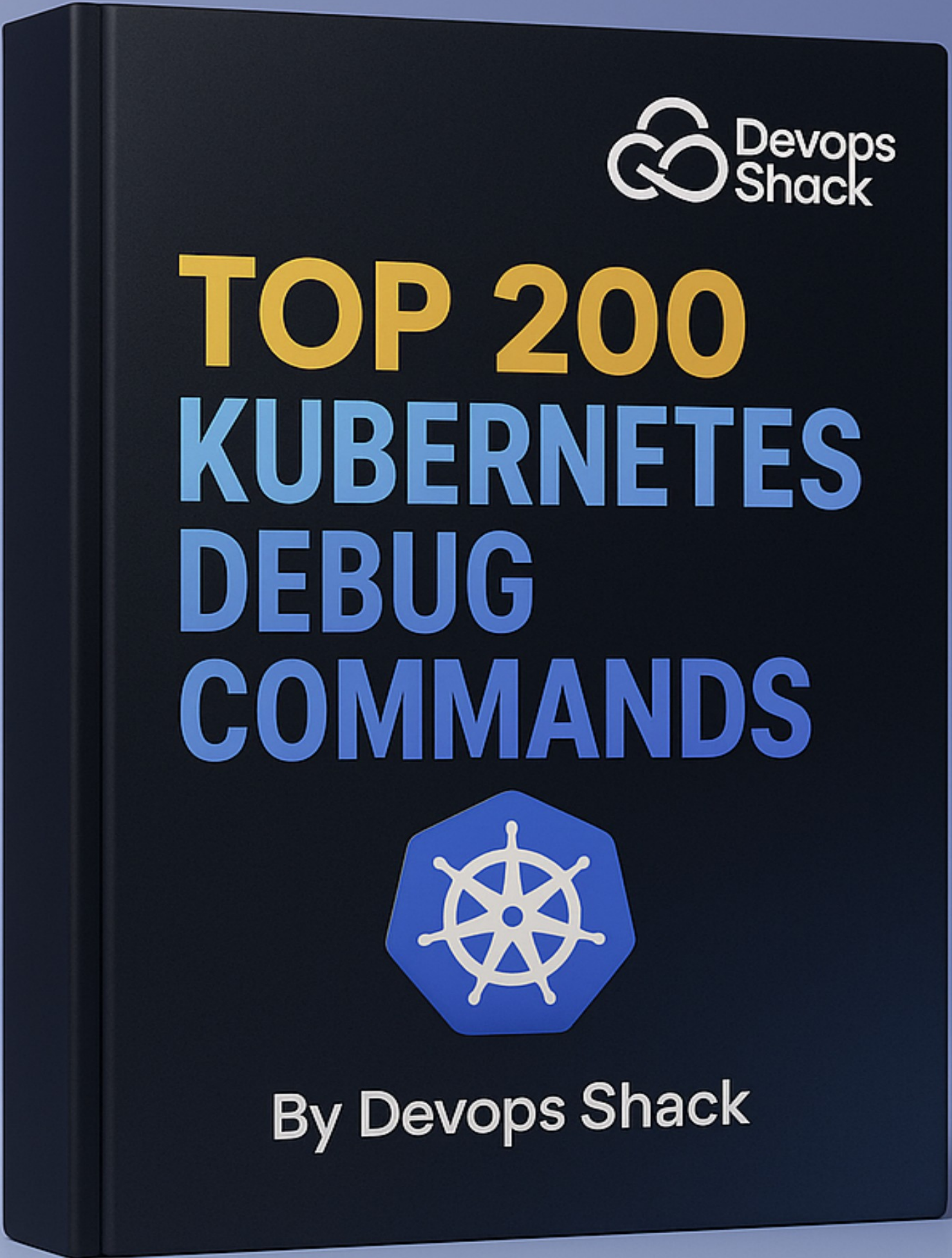

Devops
Shack

TOP 200 KUBERNETES DEBUG COMMANDS

By Devops Shack

[Click here for DevSecOps and Cloud DevOps Course](#)

DevOps Shack

Top 200 Kubernetes Debug Commands

- `kubectl version --short` — Catch client/server version skew.
- `kubectl api-resources` — Verify resource/CRD exists.
- `kubectl api-versions` — See supported API versions (deprecations).
- `kubectl config get-contexts` — Ensure you’re on the right cluster.
- `kubectl config current-context` — Print active context.
- `kubectl config use-context <ctx>` — Switch clusters quickly.
- `kubectl config view --minify -o jsonpath='{.contexts[0].context.namespace}'` — Default namespace sanity check.
- `kubectl get --raw='/livez'` — API liveness probe.
- `kubectl get --raw='/readyz?verbose'` — API readiness with failing checks.
- `kubectl get ns` — List namespaces (find your workload).
- `kubectl describe ns <ns>` — Quotas/limitRanges blocking pods.
- `kubectl get resourcequota -n <ns>` — “Exceeded quota” triage.
- `kubectl get limitrange -n <ns>` — Default CPU/mem constraints.
- `kubectl get events -n <ns> --sort-by=.lastTimestamp | tail -n 20` — Latest issues in ns.
- `kubectl get nodes -o wide` — Node statuses, versions, IPs.
- `kubectl describe node <node>` — Taints/conditions/capacity.
- `kubectl top node` — Node CPU/mem pressure (needs metrics-server).

- kubectl get pods -A --field-selector spec.nodeName=<node> — What’s on that node.
- kubectl cordon <node> — Stop scheduling to a bad node.
- kubectl drain <node> --ignore-daemonsets --delete-emptydir-data — Evict pods for maintenance.
- kubectl uncordon <node> — Return node to service.
- kubectl get node <node> -o jsonpath='{.status.addresses[*].address}' — Node IPs/hostnames.
- kubectl get node <node> -o json | jq '.status.conditions' — Scriptable node condition check.
- kubectl get pods -A --field-selector status.phase=Failed — Cluster-wide failed pods.
- kubectl get pods -A --field-selector status.phase=Pending — Scheduling backlog.
- kubectl get pods -A -o custom-columns=NS:.metadata.namespace,POD:.metadata.name,NODE:.spec.nodeName,PHASE:.status.phase — Fast overview.
- kubectl get nodes -o jsonpath='{range .items[*]}{.metadata.name}{"\t"}{.spec.taints}{"\n"}{end}' — See taints quickly.
- kubectl debug node/<node> -it --image=nicolaka/netshoot — Node-level net debug.
- kubectl get pods -n <ns> — Start pod triage here.
- kubectl get pods -n <ns> -o wide — Pod IPs/node placement.
- kubectl describe pod <pod> -n <ns> — Events/probe/image errors.
- kubectl get pod <pod> -n <ns> -o yaml — Full live manifest/status.
- kubectl get pod <pod> -n <ns> -o jsonpath='{.status.containerStatuses[*].state}' — Waiting/Running/Terminated.
- kubectl logs <pod> -n <ns> — Container logs.
- kubectl logs <pod> -c <container> -n <ns> — Target container logs.
- kubectl logs <pod> -c <container> -n <ns> --previous — CrashLoop root cause.
- kubectl logs -l app=<label> -n <ns> --tail=100 — Aggregate logs by label.
- kubectl logs <pod> -n <ns> -f — Follow logs live.

- `kubectl exec -it <pod> -n <ns> -- sh` — Shell into container.
- `kubectl cp <ns>/<pod>:/path/in/pod /tmp/local` — Pull files for analysis.
- `kubectl delete pod <pod> -n <ns> --grace-period=0 --force` — Remove stuck pod object.
- `kubectl wait --for=condition=Ready pod/<pod> -n <ns> --timeout=120s` — Gate on readiness.
- `kubectl get pod <pod> -n <ns> -o jsonpath='{.status.qosClass}'` — QoS (OOM/eviction hints).
- `kubectl get pod <pod> -n <ns> -o jsonpath='{.metadata.ownerReferences}'` — Who owns this pod (RS/Job).
- `kubectl label pod <pod> debug=true -n <ns>` — Tag for selectors.
- `kubectl annotate pod <pod> reason='investigation' -n <ns>` — Leave breadcrumbs.
- `kubectl get pod <pod> -n <ns> -o jsonpath='{.spec.affinity}'` — Affinity/anti-affinity debug.
- `kubectl get pod <pod> -n <ns> -o jsonpath='{.spec.tolerations}'` — Needs to tolerate taints?
- `kubectl get events -n <ns> --for pod/<pod>` — Pod-scoped events only.
- `kubectl get svc -n <ns>` — Services list.
- `kubectl describe svc <svc> -n <ns>` — Selector/ports/endpoints.
- `kubectl get endpoints <svc> -n <ns>` — Backing IP:port targets.
- `kubectl get ep -n <ns> -o wide` — Endpoint details (ports mismatch?).
- `kubectl port-forward svc/<svc> 8080:80 -n <ns>` — Test locally.
- `kubectl port-forward pod/<pod> 8080:8080 -n <ns>` — Direct to pod.
- `kubectl get svc <svc> -n <ns> -o jsonpath='{.spec.type}'` — ClusterIP/NodePort/LB.
- `kubectl get svc <svc> -n <ns> -o jsonpath='{.spec.sessionAffinity}'` — Sticky sessions?
- `kubectl get endpoints <svc> -n <ns> -o jsonpath='{.subsets[*].addresses[*].targetRef.name}'` — Pod names behind service.
- `kubectl get service <svc> -n <ns> -o yaml | yq '.spec.ports'` — Validate port/targetPort.
- `kubectl get ingress -n <ns>` — Ingress list.

- kubectl describe ingress <ing> -n <ns> — Rules, class, TLS, events.
- kubectl get ing <ing> -n <ns> -o yaml — Check annotations/class.
- kubectl get ingressclass — Is the controller class present?
- kubectl get gateway,httproute -n <ns> — If using Gateway API.
- kubectl describe httproute <route> -n <ns> — Path/host matching issues.
- kubectl get certificate -n <ns> — cert-manager certs status.
- kubectl describe challenge -n <ns> — ACME challenges debug.
- kubectl get svc kube-dns -n kube-system -o yaml — CoreDNS service.
- kubectl get configmap coredns -n kube-system -o yaml — CoreDNS config (stubDomains, rewrites).
- kubectl -n kube-system get pods -l k8s-app=kube-dns -o wide — CoreDNS pods healthy/where.
- kubectl -n kube-system logs -l k8s-app=kube-dns — DNS errors/timeouts.
- kubectl exec -it <pod> -n <ns> -- nslookup <svc> — In-pod DNS resolution.
- kubectl exec -it <pod> -n <ns> -- dig <svc> +short — FQDN→IP mapping (if dig present).
- kubectl exec -it <pod> -n <ns> -- cat /etc/resolv.conf — Search domains & DNS policy.
- kubectl exec -it <pod> -n <ns> -- curl -sv http://<svc>:<port>/health — HTTP reachability.
- kubectl exec -it <pod> -n <ns> -- ss -tulpn — Sockets/listeners check.
- kubectl exec -it <pod> -n <ns> -- netstat -plnt — Legacy sockets list.
- kubectl exec -it <pod> -n <ns> -- ip route — Routing table in pod.
- kubectl exec -it <pod> -n <ns> -- tcpdump -i any port <p> -c 50 — Packet capture (if permitted).
- kubectl get deploy -n <ns> — Find owning deployment.
- kubectl describe deploy <dep> -n <ns> — Conditions/events/strategy.
- kubectl rollout status deploy/<dep> -n <ns> — Watch rollout complete/fail.

- `kubectl rollout history deploy/<dep> -n <ns>` — What changed last time.
- `kubectl rollout undo deploy/<dep> -n <ns> --to-revision=<n>` — Fast rollback.
- `kubectl set image deploy/<dep> <ctr>=<img>:<tag> -n <ns>` — Hotfix image/tag.
- `kubectl scale deploy/<dep> --replicas=0 -n <ns>` — Quarantine noisy workload.
- `kubectl set env deploy/<dep> KEY=VALUE -n <ns>` — Flip feature flag/env.
- `kubectl diff -f deploy.yaml` — Live vs file server-side diff.
- `kubectl apply -f deploy.yaml --server-side --dry-run=server -o yaml` — Validate change without mutating.
- `kubectl get rs -n <ns>` — ReplicaSets (orphaned?)
- `kubectl describe rs <rs> -n <ns>` — Why replicas not created.
- `kubectl get ds -n <ns> -o wide` — DaemonSets per node.
- `kubectl describe ds <ds> -n <ns>` — Node selectors/taints issues.
- `kubectl get sts -n <ns> -o wide` — StatefulSet & ordinals.
- `kubectl describe sts <sts> -n <ns>` — Stuck ordinal/PVC bindings.
- `kubectl get jobs -n <ns>` — Job completions/failures.
- `kubectl describe job <job> -n <ns>` — Backoff limits & pods.
- `kubectl logs job/<job> -n <ns> --all-containers` — Consolidated job output.
- `kubectl get cj -n <ns>` — CronJobs schedule/last run.
- `kubectl describe cj <cron> -n <ns>` — CronJob details (missed runs, concurrency).
- `kubectl create job --from=cronjob/<cron> manual-<ts> -n <ns>` — Reproduce a CronJob run.
- `kubectl get pvc -n <ns>` — List claims; spot Pending/Bound.
- `kubectl describe pvc <pvc> -n <ns>` — Events: binding/class/size issues.
- `kubectl get pv` — PV capacity/reclaim policy/phase.
- `kubectl describe pv <pv>` — Node affinity/attach errors.

- kubectl get sc — StorageClasses; find default.
- kubectl describe sc <sc> — Provisioner params/timeouts.
- kubectl get volumeattachment — CSI attach/detach objects.
- kubectl describe volumeattachment <name> — Why attach is stuck.
- kubectl exec -it <pod> -n <ns> -- df -h — In-container disk fullness.
- kubectl exec -it <pod> -n <ns> -- mount — Mount paths & types.
- kubectl get events -n <ns> --field-selector involvedObject.kind=PersistentVolumeClaim — PVC-only events.
- kubectl get events -A --sort-by=.lastTimestamp | tail -n 50 — Latest cluster incidents.
- kubectl get events --field-selector reason=FailedScheduling -A — Scheduling denials.
- kubectl get events -A --field-selector reason=BackOff — CrashLoop/BackOff storms.
- kubectl get events -A --field-selector reason=Killing — Pods killed due to updates/eviction.
- kubectl get events -n <ns> --since=30m — Zoom into incident window.
- kubectl get lease -A — Leader elections flapping.
- kubectl get lease -n kube-system — Controller/scheduler leadership.
- kubectl top pod -n <ns> — Hot pods at a glance (needs metrics-server).
- kubectl top pod -l app=<label> -n <ns> — Compare replicas of same app.
- kubectl get hpa -n <ns> — Autoscalers present?
- kubectl describe hpa <hpa> -n <ns> — Metrics/desired replicas/last scale.
- kubectl get pods -n <ns> -o jsonpath='{range .items[*]}{.metadata.name}{"\t"}{.spec.containers[*].resources}{"\n"}{end}' — Requests/limits audit.
- kubectl describe pod <pod> -n <ns> | grep -i oom — OOMKilled traces.
- kubectl get events -A --field-selector reason=Evicted — Node pressure evictions.
- kubectl get rs/<rs> -n <ns> -o jsonpath='{.status.availableReplicas}' — RS availability.

- kubectl get deploy/<dep> -n <ns> -o jsonpath='{.status.conditions}' — Blocked rollout reason.
- kubectl rollout restart deploy/<dep> -n <ns> — Pick up config/secret changes.
- kubectl get pod <pod> -n <ns> -o jsonpath='{.spec.nodeName}' — Which node hosts it.
- kubectl describe pod <pod> -n <ns> | sed -n '/Events:/,$p' — Only events section.
- kubectl get nodes --show-labels — Node labels for selectors.
- kubectl get pod <pod> -n <ns> -o jsonpath='{.spec.nodeSelector}' — Selector/label mismatch.
- kubectl taint nodes <node> key=value:NoSchedule — Quarantine node/steer placement.
- kubectl taint nodes <node> key=value:NoSchedule- — Remove taint.
- kubectl describe priorityclass — Preemption/priority factors.
- kubectl get scheduling.k8s.io/priorityclass -o yaml — Cluster-wide priorities.
- kubectl get pod <pod> -n <ns> -o jsonpath='{.spec.affinity}' — Affinity/anti-affinity rules.
- kubectl get pod <pod> -n <ns> -o jsonpath='{.spec.tolerations}' — Toleration confirms landing on tainted nodes.
- kubectl auth can-i list pods -n <ns> --as <user> — Simulate RBAC.
- kubectl get role,rolebinding -n <ns> — Who can do what in ns.
- kubectl describe rolebinding -n <ns> — Subjects bound in namespace.
- kubectl describe clusterrolebinding <name> — Wide permissions audit.
- kubectl get sa -n <ns> — SAs used by workloads.
- kubectl describe sa <sa> -n <ns> — Tokens, imagePullSecrets.
- kubectl get secret -n <ns> — Required secrets present?
- kubectl describe secret <name> -n <ns> — Types/annotations/owners (not values).
- kubectl auth can-i get secrets --as <user> -n <ns> — Confirm secret visibility.
- kubectl auth reconcile -f rbac.yaml --dry-run=client -o yaml — Plan safe RBAC changes.
- kubectl get pod <pod> -n <ns> -o jsonpath='{.status.podIP}' — Extract pod IP only.

- kubectl get svc <svc> -n <ns> -o jsonpath='{.spec.clusterIP}' — ClusterIP only.
- kubectl get ing <ing> -n <ns> -o jsonpath='{.status.loadBalancer.ingress[0].ip}' — LB IP.
- kubectl get svc <svc> -n <ns> -o jsonpath='{.spec.externalTrafficPolicy}' — Source IP preservation.
- kubectl get svc <svc> -n <ns> -o jsonpath='{.status.loadBalancer.ingress[*].hostname}' — Cloud LB hostnames.
- kubectl get pods -n <ns> -o custom-columns=NAME:.metadata.name,READY:.status.containerStatuses[*].ready,RESTARTS:.status.containerStatuses[*].restartCount — At-a-glance health.
- kubectl get pods -A -l app=<label> -o name — Names only (for piping).
- kubectl get pods -n <ns> --show-labels — Inline labels for selector debug.
- kubectl get pods -n <ns> -l 'app in (a,b)' — Label-set selection.
- kubectl get pods --field-selector spec.nodeName=<node> -n <ns> — Pods bound to a node.
- kubectl debug pod/<pod> -n <ns> -it --image=busybox --target=<container> — Ephemeral debug container.
- kubectl debug -it --image=busybox --attach=false --share-processes --copy-to=dbg-<pod> pod/<pod> -n <ns> — Copy with shared PID ns.
- kubectl get pod <pod> -n <ns> -o jsonpath='{.spec.ephemeralContainers}' — Audit ephemeral containers.
- kubectl delete pod dbg-<pod> -n <ns> — Clean up debug copy.
- kubectl exec -it <pod> -n <ns> -- pstree -al — Process tree visibility.
- kubectl -n kube-system logs -l component=kube-scheduler --tail=200 — Scheduler logs (label may vary).
- kubectl -n kube-system logs -l component=kube-controller-manager --tail=200 — Controller-manager logs.
- kubectl -n kube-system get pods -o wide — System pods/node placement.
- kubectl -n kube-system describe pod <cp-pod> — Limits/args/env of control-plane pod.
- kubectl -n kube-system get events --sort-by=.lastTimestamp | tail -n 30 — Recent control-plane events.

- kubectl get node <node> -o jsonpath='{.status.nodeInfo.containerRuntimeVersion}' — CRI & version.

- kubectl debug node/<node> -it --image=nicolaka/netshoot -- bash — Node network namespace.

- crictl ps -a — List containers via CRI (run **on node**).

- crictl logs <container-id> — Logs via CRI when kubectl can’t.

- crictl inspect <container-id> | jq '.status.exitCode,.status.reason' — Exit metadata.

- journalctl -u kubelet --since '1 hour ago' — Kubelet log stream (on node).

- ls /var/log/containers | grep <pod> — Find container logs symlinks (on node).

- crictl images | grep <repo> — Confirm image cached (on node).

- sudo ss -plnt | grep kube-proxy — kube-proxy listening (on node).

- iptables -S | grep KUBE- — iptables mode rules (on node).

- kubectl get endpointslices -n <ns> — EndpointSlice health (modern discovery).

- kubectl get endpointslices.discovery.k8s.io -n <ns> -o wide — Hints/topology.

- kubectl get mutatingwebhookconfigurations,validatingwebhookconfigurations — Admission webhooks present?

- kubectl describe validatingwebhookconfiguration <name> — Rules, failurePolicy, timeouts.

- kubectl apply -f <file> --dry-run=client -o yaml — Client-side render check.

- kubectl diff -f <file> — Server diff (live vs desired).

- kubectl apply -f <file> --server-side --dry-run=server -o yaml — Server schema/validation check.

- kubectl wait --for=condition=Available deploy/<dep> -n <ns> --timeout=90s — Block until ready.

- kubectl set image deploy/<dep> *=<image>:<tag> --record -n <ns> — Update all containers + record.

- kubectl annotate deploy/<dep> kubernetes.io/change-cause='hotfix' -n <ns> — Human-friendly rollout history.

- kubectl get secret <name> -n <ns> -o jsonpath='{.type}' — Opaque vs dockerconfigjson.

- kubectl get pod <pod> -n <ns> -o jsonpath='{.spec.imagePullSecrets}' — Pull secret wired?
- kubectl get cm -n <ns> — ConfigMap inventory.
- kubectl describe cm <cm> -n <ns> — Config contents/metadata.
- kubectl get deploy -n <ns> -o jsonpath='{range .items[*]}{.metadata.name}{"\t"}{.spec.template.spec.containers[*].image}{"\n"}{end}' — Image audit across deployments.
- kubectl set resources deploy/<dep> -n <ns> --limits=cpu=500m,memory=512Mi --requests=cpu=250m,memory=256Mi — Hot adjust resources.
- kubectl get crd | head — CRDs exist?
- kubectl describe <crd-kind> <name> -n <ns> — CRD instance detail.
- kubectl get cm -n kube-system kubeadm-config -o yaml — Kubeadm cluster config reference.
- kubectl get pods -A -o custom-columns=NS:.metadata.namespace,POD:.metadata.name,PHASE:.status.phase,RESTARTS:.status.containerStatuses[*].restartCount | column -t — Cluster-wide health snapshot.