Runbook — Change Kubernetes Pod Subnet from 192.168.0.0/16 to 172.22.0.0/16 (Flannel + kubeadm)

DATE
16 Oct 2025

ENVIRONMENT
kubeadm v1.30, Ubuntu 24.04, Flannel v0.27 (VXLAN), two nodes: master (control-plane) and worker.

OBJECTIVE
Migrate Pod network from 192.168.0.0/16 to 172.22.0.0/16 without rebuilding the cluster.

SUMMARY OF APPROACH
1) Update cluster CIDR in kube-controller-manager, kube-proxy, and Flannel.
2) Reissue PodCIDRs by deleting Node objects while kubelet is stopped (to clear old 192.168.x assignments).
3) Wipe local CNI state; start kubelet; let NodeIPAM assign new 172.22.x/24 per node.
4) Restart Flannel and roll workloads to pick up new Pod IPs.

PRE-CHECKS
• Confirm target Pod CIDR is available: 172.22.0.0/16.
• Update any firewalls/VPN allowlists to include 172.22.0.0/16 before the change.
• Ensure Deployments have readiness/liveness probes and PodDisruptionBudgets as applicable.

CHANGE STEPS (EXECUTED)

A) UPDATE CONTROL PLANE AND CNI CONFIGS
1. kube-controller-manager: set cluster CIDR to 172.22.0.0/16
   File: /etc/kubernetes/manifests/kube-controller-manager.yaml
   Change: --cluster-cidr=172.22.0.0/16
   (kubelet auto-restarts the static pod)
2. kube-proxy: update clusterCIDR
   kubectl -n kube-system get cm kube-proxy -o yaml | sed 's/clusterCIDR: .*/clusterCIDR: "172.22.0.0\/16"/' | kubectl apply -f -
   kubectl -n kube-system rollout restart ds kube-proxy
3. Flannel ConfigMap: set net-conf.json → "Network": "172.22.0.0/16"
   kubectl -n kube-flannel edit cm kube-flannel-cfg
   kubectl -n kube-flannel rollout restart ds kube-flannel-ds

B) REISSUE PODCIDRS AND CLEAN NODE CNI STATE
4. On WORKER node: stop kubelet and wipe old CNI state
   sudo systemctl stop kubelet
   sudo rm -rf /var/lib/cni/networks/*
   sudo rm -f /etc/cni/net.d/10-flannel.conflist
   sudo ip link delete cni0 2>/dev/null || true
   sudo ip link delete flannel.1 2>/dev/null || true
5. From master: delete worker Node object
   kubectl delete node worker
6. Start kubelet on WORKER
   sudo systemctl start kubelet

```
   (Node re-registers and will later get a 172.22.x/24 PodCIDR)
7. From master: delete master Node object (clears old PodCIDR on control-plane)
   kubectl delete node master
8. On MASTER OS: wipe CNI cache then restart kubelet
   sudo systemctl stop kubelet
   sudo rm -rf /var/lib/cni/networks/*
   sudo rm -f  /etc/cni/net.d/10-flannel.conflist
   sudo ip link delete cni0 2>/dev/null || true
   sudo ip link delete flannel.1 2>/dev/null || true
   sudo systemctl start kubelet
9. Bounce controller-manager once (ensures node-ipam starts cleanly)
   kubectl -n kube-system delete pod kube-controller-manager-master
   (recreated automatically by static manifest)

C) VERIFY NODE PODCIDRS AND FLANNEL HEALTH
10. Watch nodes receive new PodCIDRs
    kubectl get node -o custom-
columns=NAME:.metadata.name,PODCIDR:.spec.podCIDR,PODCIDRS:.spec.podCIDRs -w
    (Expect 172.22.X.0/24 per node)
11. Ensure Flannel pods are Running
    kubectl -n kube-flannel get pods -o wide

D) ROLL WORKLOADS TO PICK UP NEW POD IPS
12. Restart deployments
    kubectl rollout restart deploy -A
13. Confirm pods have 172.22.x addresses
    kubectl get pods -A -o wide | awk 'NR>1{print $6}' | grep '^172\.22\.'
14. Connectivity test from a fresh pod
    kubectl -n default run netcheck --rm -it --image=alpine:3.19 -- sh
    apk add --no-cache busybox-extras
    ip -4 a    # should show 172.22.x
    nc -vz 192.168.1.100 5432   # adjust port as needed

POST-CHANGE
• Remove old 192.168.0.0/16 references from firewalls/VPNs after validation.
• If the control-plane was temporarily schedulable, re-taint:
  kubectl taint nodes master node-role.kubernetes.io/control-plane=NoSchedule:NoSchedule --overwrite
• Monitor logs/metrics and verify cross-node pod↔pod, pod↔Service, and pod↔external connectivity.

ROLLBACK PLAN (IF NEEDED)
1) Revert kube-controller-manager --cluster-cidr, kube-proxy clusterCIDR, and Flannel net-conf.json
back to 192.168.0.0/16.
2) Stop kubelet, wipe CNI, delete Node objects, start kubelet to reissue old PodCIDRs.
3) Restart Flannel and confirm pods return to 192.168.x.x.
```