What is an exposed kubelet?

The kubelet is the agent that runs on every Kubernetes node and manages its pods. Its API listens on port 10250, and if it allows anonymous access (or is reachable from where it should not be), an attacker can list pods and execute commands inside them without credentials, harvesting secrets and service account tokens. An exposed, unauthenticated kubelet is a direct route from network access to running code in workloads. Lock it down with authentication and authorization.

HOW IT WORKS

01 The abuse and payload

Against a kubelet that permits anonymous access on 10250:

- List the pods on the node: `curl -sk https://NODE:10250/pods`
- Execute a command inside a chosen pod/container (the kubelet `run/exec` endpoint), e.g. read its mounted service account token at `/var/run/secrets/kubernetes.io/serviceaccount/token`.
- Use that token against the API server to expand access across the cluster (see service account token).

Documented techniques shown for defenders.

> ANONYMOUS = EXEC IN ANY POD
>
> *An anonymously reachable kubelet on 10250 lets an attacker run commands inside the node's pods with no credentials, then steal their tokens. It is a fast path from the network to the cluster.*

HOW TO DEFEND

- Disable anonymous auth on the kubelet (`--anonymous-auth=false`) and require authentication.
- Enable authorization (`--authorization-mode=Webhook`) so even authenticated callers are checked.
- Restrict network access to 10250 so only the control plane can reach it.
- Rotate and scope service account tokens so a stolen one is limited.
- Scan nodes for exposed kubelet ports and test the cluster for this path.

SOURCES

[1] Kubernetes docs: Kubelet authentication/authorization

[2] MITRE ATT&CK: Containers Matrix

[3] NIST SP 800-190 Application Container Security Guide

Find the container escape paths before an attacker does.

securelayer7.net/learn/containers/what-is-an-exposed-kubelet

Open online