What is a service account token?

A Kubernetes service account token is a credential mounted into a pod (by default at `/var/run/secrets/kubernetes.io/serviceaccount/token`) that the pod uses to authenticate to the API server. Its power is whatever RBAC grants that service account. When an attacker gets code execution in a pod, the token is the first thing they steal: they read it from disk and use it to query and act on the cluster. Over-permissioned tokens and unnecessary automounting are the core risk.

HOW IT WORKS

01 The abuse and payload

After getting code execution in a pod, the attacker grabs and uses the token:

- Read it: `cat /var/run/secrets/kubernetes.io/serviceaccount/token` (plus the namespace and CA in the same directory).
- Use it against the API server: `kubectl --token=$TOK --server=https://API auth can-i --list` to enumerate what it can do.
- If RBAC is broad, list/read secrets, create pods, or otherwise escalate toward cluster control.

Documented techniques shown for defenders.

> THE POD'S FIRST KEY
>
> *The service account token is the first credential an attacker finds in a compromised pod. Its danger is set entirely by RBAC, scope it tightly and do not mount it where it is not needed.*

HOW TO DEFEND

- Disable token automount where a pod does not need the API (`automountServiceAccountToken: false`).
- Scope each service account with least-privilege RBAC; never bind cluster-admin.
- Use short-lived, audience-bound projected tokens rather than long-lived ones.
- Give each workload its own service account, not a shared or default one.
- Detect unexpected API calls from pod tokens and test the cluster for token-based escalation.

SOURCES

[1] Kubernetes docs: Configure service accounts

[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-a-kubernetes-service-account-token

Open online