SELF-HOST > INSTALLATIONS- & BEREITSTELLUNGSANLEITUNGEN >

# Kubernetes Service Accounts

## Kubernetes Service Accounts

Kubernetes service accounts can be used to apply specific security contexts to specific pods. This can be useful, for example, in scenarios where you need to run your Bitwarden server in rootless mode, as the included SQL container requires elevated permissions.

Once you've created and configured your service account with the desired permissions, change any of the pod service account designations (for example, `database.podServiceAccount`) in your `my-values.yaml` file. For example, a `my-values.yaml` with `component.admin.podServiceAccount` assigned a service account named `bitwarden-sa` should look like the following:

```Bash
component:
  # The Admin component
  admin:
    # Additional deployment labels
    labels: {}
    # Image name, tag, and pull policy
    image:
      name: ghcr.io/bitwarden/admin
    resources:
      requests:
        memory: "64Mi"
        cpu: "50m"
      limits:
        memory: "128Mi"
        cpu: "100m"
    securityContext:
    podServiceAccount: bitwarden-sa
```

Pods that are eligible for service account designation include:

- `component.admin.podServiceAccount`
- `component.api.podServiceAccount`
- `component.attachments.podServiceAccount`
- `component.events.podServiceAccount`
- `component.icons.podServiceAccount`
- `component.identity.podServiceAccount`
- `component.notifications.podServiceAccount`

- `component.scim.podServiceAccount`
- `component.sso.podServiceAccount`
- `component.web.podServiceAccount`
- `database.podServiceAccount`