LOGZILLA DOCUMENTATION

Kubernetes Prerequisites

Kubernetes cluster requirements for a LogZilla deployment, covering LoadBalancer services, Ingress controllers, StorageClasses, image registries, and namespaces

Kubernetes Deployment Overview · Generated June 12, 2026 · logzilla.ai/docs/kubernetes-deployment/prerequisites

Cluster requirements

- LoadBalancer Service support for external syslog and HTTP ingest.
- An Ingress controller compatible with `networking.k8s.io/v1`.
- A default `StorageClass` or a named `StorageClass` for PersistentVolumeClaims.
- Sufficient node capacity for the requested CPU and memory in manifests.

Registry and images

- Pull access to LogZilla images.
- Use the `stable` tag by default for production. If a release pinning policy exists, use a specific tag (for example, `vX.Y.Z`).
- Third-party images (PostgreSQL, Redis, InfluxDB, Grafana) use stable tags in the examples.

Namespace

If using a dedicated namespace:

```
kubectl create namespace logzilla
# add -n logzilla to subsequent kubectl commands
```

Secrets: stringData vs data

- Examples in this section use `stringData` for Secrets to keep manifests readable; Kubernetes encodes them to base64 on apply.
- If using `data:` instead, base64-encode values first. Example:

```
printf '%s' 'your_token_here' | base64
```

StorageClass notes

- In some manifests, a `storageClassName` is referenced (for example, `premium-rwo`). Replace it with the cluster's StorageClass or remove the field if the default StorageClass is desired.

Order of deployment

Backing services (PostgreSQL, Redis, InfluxDB, optional Grafana)
Common ConfigMaps and Secrets
LogZilla modules (Storage, Query, API, Ingest, Front)
Ingress and external exposure
Refer to each page in this section for copy-and-paste manifests.