QSAN Official Document

DevOps Consolidation Guide

Application Note

July 2026

ANNOUNCEMENT

Copyright

© Copyright 2026 QSAN Technology, Inc. All rights reserved. No part of this document may be reproduced or transmitted without written permission from QSAN Technology, Inc.

QSAN believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

Trademarks

- QSAN, the QSAN logo, and QSAN.com are trademarks or registered trademarks of QSAN Technology, Inc.
- Microsoft, Windows, Windows Server, and Hyper-V are trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries.
- Linux is a trademark of Linus Torvalds in the United States and/or other countries.
- UNIX is a registered trademark of The Open Group in the United States and other countries.
- Mac and OS X are trademarks of Apple Inc., registered in the U.S. and other countries.
- Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.
- VMware, ESXi, and vSphere are registered trademarks or trademarks of VMware, Inc. in the United States and/or other countries.
- Citrix and Xen are registered trademarks or trademarks of Citrix Systems, Inc. in the United States and/or other countries.
- Other trademarks and trade names used in this document to refer to either the entities claiming the marks and names or their products are the property of their respective owners.

TABLE OF CONTENTS

Announcement i
Notices v
Preface vi
Technical Support vi
Information, Tip, and Caution vi
1. Introduction to DevOps 1
2. Configuration Settings 2
2.1. Environment and Topology 2
2.2. Deployment and Configuration Steps 3
3. Conclusion 6
4. Appendix 7
4.1. Apply To 7

©2026 QSAN Technology, Inc. All rights reserved.

FIGURES

Figure 2-1 Demonstration Topology 3

Announcement
©2026 QSAN Technology, Inc. All rights reserved.
www.qsan.com

TABLES

Table 2-1 KSM System Resource Configuration 4

Announcement
©2026 QSAN Technology, Inc. All rights reserved.
www.qsan.com

NOTICES

Information contained in this document has been reviewed for accuracy. But it could include typographical errors or technical inaccuracies. Changes are made to the document periodically. These changes will be incorporated in new editions of the publication. QSAN may make improvements or changes in the products. All features, functionality, and product specifications are subject to change without prior notice or obligation. All statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

©2026 QSAN Technology, Inc. All rights reserved.

PREFACE

Technical Support

Do you have any questions or need help trouble-shooting a problem? Please contact QSAN Support, we will reply to you as soon as possible.

- Via the Web: https://www.qsan.com/technical_support
- Via Telephone: +886-2-77206355
- (Service hours: 09:30 - 18:00, Monday - Friday, UTC+8)
- Via Skype Chat, Skype ID: qsan.support
- (Service hours: 09:30 - 02:00, Monday - Friday, UTC+8, Summer time: 09:30 - 01:00)
- Via Email: support@qsan.com

Information, Tip, and Caution

This document uses the following symbols to draw attention to important safety and operational information.



INFORMATION

INFORMATION provides useful knowledge, definition, or terminology for reference.



TIP

TIP provides helpful suggestions for performing tasks more effectively.

©2026 QSAN Technology, Inc. All rights reserved.



CAUTION

CAUTION indicates that failure to take a specified action could result in damage to the system.

©2026 QSAN Technology, Inc. All rights reserved.

1. INTRODUCTION TO DEVOPS

Modern software development requires rapid iteration, continuous deployment, and reliable automation. As teams scale and microservices architectures become mainstream, organizations often face challenges in standardizing their CI/CD pipelines, maintaining security compliance, and controlling cloud-based DevOps costs.

QSAN KSM (Kubernetes Server Management) provides an enterprise-grade container foundation for building a fully private, high-performance DevOps platform. By deploying GitLab on top of a KSM cluster—with integrated runners, container registry, and S3-compatible object storage—enterprises can achieve a consistent, secure, and scalable CI/CD environment that supports company-wide development workflows.

This Application Note outlines how to build a cloud-native DevOps ecosystem using KSM × GitLab, and demonstrates the architectural components, deployment workflow, and operational benefits.

©2026 QSAN Technology, Inc. All rights reserved.

2. CONFIGURATION SETTINGS

2.1. Environment and Topology

Demonstration Environment

- Server
  - Model: KS2204-8C
    CPU: 8 cores
    Memory: 16 GB
    OS: KSM 1.0.0
- Object Storage
  - Model: XN8124D x 2
    Memory: 16 GB per controller
- GitLab Component
  - **GitLab (Helm Deployment)**
    GitLab web services, Sidekiq, Gitaly, Redis, and PostgreSQL.
  - **GitLab Runners**
    Pod-per-job execution model for highly parallel CI workloads.
  - **Private Container Registry**
    On-premise storage for Docker images with access control.
  - **Object Storage (MinIO or S3)**
    Storage for artifacts, LFS objects, packages, and dependency cache.
  - **Ingress & TLS**
    HTTPS access for GitLab and internal components.

©2026 QSAN Technology, Inc. All rights reserved.

Demonstration Topology

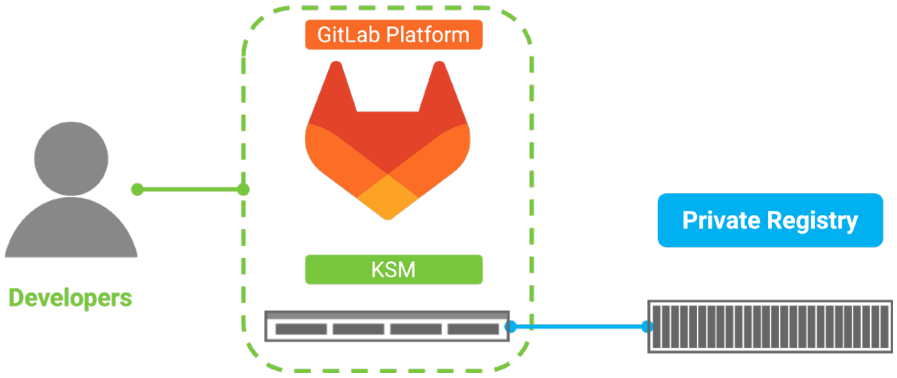


Figure 2-1 Demonstration Topology

2.2. Deployment and Configuration Steps

Step 1. Prepare KSM Cluster

1. Enable RBAC
2. Allocate persistent volumes for GitLab, registry, and PostgreSQL
3. Configure worker node resources for CI workloads

CAUTION

Initial sizing requires at least 50GB for PostgreSQL and 100GB for MinIO artifacts.

Step 2. Configure Object Storage

1. Create MinIO / S3 buckets for:
   - Artifacts
   - LFS
   - Packages
   - Registry

©2026 QSAN Technology, Inc. All rights reserved.
www.qsan.com
Official Document

2. Generate access keys
3. Apply bucket retention lifecycle policies

Step 3. Deploy GitLab via Helm

1. Add GitLab Helm repository
2. Prepare custom *values.yaml*
3. Apply resource limits to each GitLab service
4. Configure persistent volumes for Gitaly and Registry
5. Enable Ingress with TLS termination

Step 4. Configure GitLab Runners

1. Install GitLab Runner Helm chart
2. Configure pod templates for job isolation
3. Apply resource requests/limits:

Table 2-1 KSM System Resource Configuration

JOB TYPE	CPU	MEMORY
Standard	0.1 core	0.1 GB
Build	2 cores	4 GB
Integration Test	4 cores	8 GB



INFORMATION

KSM will automatically scale pods based on available compute resources.

Step 5. Configure Container Registry

1. Configure the registry host, storage path, and access credentials
2. Integrate vulnerability scanning
3. Apply registry cleanup policies (tag expiration, artifact retention)

©2026 QSAN Technology, Inc. All rights reserved.

Step 6. Configure Access Control

1. Integrate LDAP or SSO
2. Apply namespace-level RBAC
3. Configure Code Owners and protected branches
4. Define merge approval rules

Step 7. Build CI/CD Pipeline

1. Establish standardized stages:
   - **Build** – Compile and generate artifacts
   - **Test** – Unit, integration, security scanning
   - **Deploy** – Multi-environment promotion
2. Create shared YAML templates for consistency
3. Configure artifact passing and caching
4. Apply approval-based promotion workflows

©2026 QSAN Technology, Inc. All rights reserved.

3. CONCLUSION

Deploying GitLab on KSM enables enterprises to build a fully private, scalable, and secure cloud-native DevOps platform.

Through the use of container-based runners, private registries, centralized object storage, and robust RBAC integration, organizations can standardize CI/CD workflows, reduce development cycle times, enhance governance, and maintain predictable infrastructure costs.

KSM empowers teams to focus on software innovation rather than infrastructure operations—delivering faster, safer, and more reliable applications.

©2026 QSAN Technology, Inc. All rights reserved.

4. APPENDIX

4.1. Apply To

- KSM firmware 1.0.0 and later

©2026 QSAN Technology, Inc. All rights reserved.