QBR Summary: Scalable Web App Deployment on DigitalOcean Kubernetes (DOKS)

Engagement Information

Customer: Fictional Industries

TAM: Srish Sriram

Engagement Type: Technical Assessment

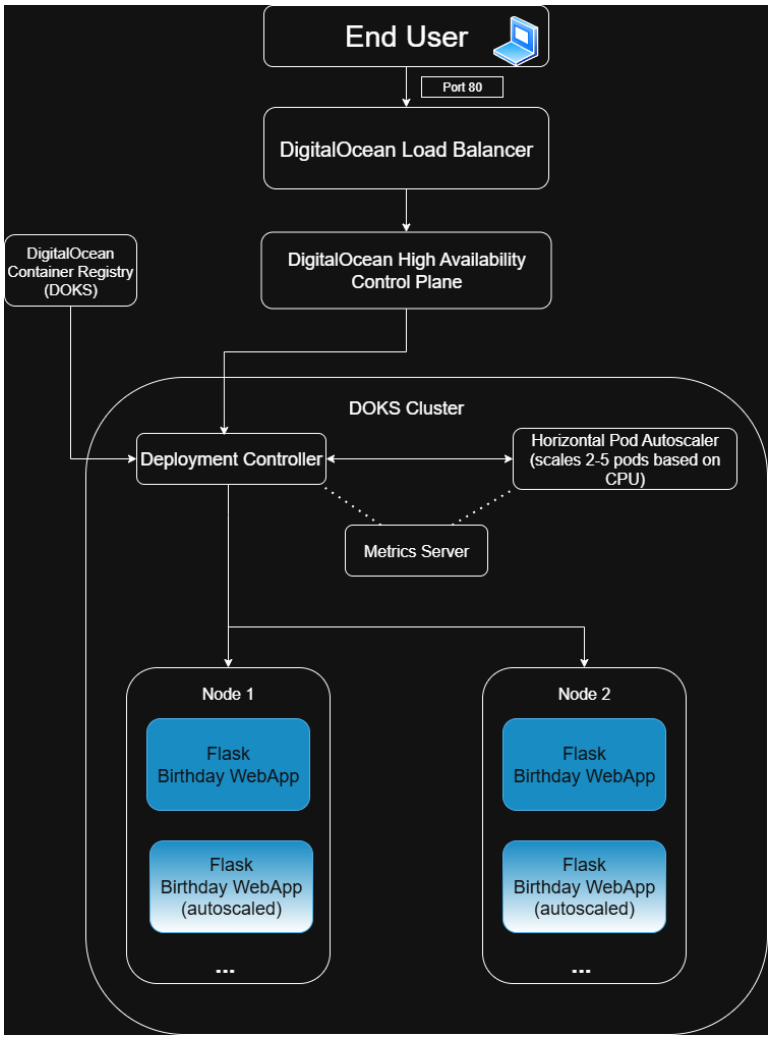
Use Case: Deploying a lightweight, scalable, and cost-efficient web application using DOKS

Executive Summary

We deployed a containerized Flask-based web application on DOKS to simulate a SaaS environment. This mockup application tracks user birthdays and displays them on a user interface. The architecture created prioritizes:

- **Scalability**, via horizontal pod autoscaling
- **Performance**, through scaling responsive to CPU loads
- **Reliability**, using a multi-node pool and high-availability control pane components
- **Cost optimization**, through efficient resource sizing and minimal idle compute through smart scaling

Current Architecture



The diagram above details the flow of system components:

- **App:** Python Flask web frontend
- **Hosting:** DOKS cluster (2-node pool)
- **Image Source:** DigitalOcean Container Registry (DOCR)
- **Networking:** DigitalOcean Load Balancer
- **Scaling:** Horizontal Pod Autoscaler (2-5 replicas based on CPU load)
- **Control Plane**: High-availability configured and managed by DigitalOcean

Performance Observations

- Horizonal autoscaling scaled from 2 to 5 pods within 30 seconds under simulated CPU load
- Load balancer IP maintained 100% uptime during testing
- Response time remained stable throughout increased simulated traffic
- Node spread was balanced across both nodes

Cost Comparison and Analysis

A direct cost analysis shows about 15% savings on technical components alone:

Component	**DOKS Managed**	**Self-Managed Kubernetes**
Worker Nodes	2 x $24 = $48	2 x $24 = $48
High-Availability Control Planes	$40	2 x $24 = $38
Load Balancer	$12	$15
Monitoring/Telemetry	$0	$5
Networking	$0	$10
Total Monthly Costs	$100/month	$115/month

Moreover, there are:

- **Reduced Engineering Overhead:** No engineering hours spent on cluster setup or management, no time spent on manual upgrades, built-in high-availability and fault tolerance, and preconfigured networking solutions.
    - Time saved: ~10-20 hours per month of DevOps effort
    - Value: $1000+
- **Fewer Production Risks:** Built in control plane and node high-availability, tested metric servers and autoscalers, simple to implement security practices, and built-in monitoring and logging.
    - Time saved: ~5 hours/month
    - Value: $300-$500/month

Risks and Mitigation Strategies

- **Risk:** Load balancer costs scale with number of services
    - **Mitigation:** Use a shared ingress with multiple host rules
- **Risk:** HPA scales slowly under heavy load

    - **Mitigation:** Set minimum replicas to 2 to prevent cold-start latency
- **Risk:** Pod restart delays due to large image size
    - **Mitigation:** Use multi-stage images to reduce file sizes

Conclusion

This deployment showcases a scalable and cost-conscious approach to deploying Kubernetes-based architecture using DigitalOcean. This system is designed to support fluctuating degrees of traffic for a growing Saas product. While this application is a mockup and not intended for full production usage, with some minor feature implementation (such as ingresses and probes) it can be easily made production-grade.