Project 2: Loki - Centralized Log Management

Grafana Cloud Dashboard for Application Log Aggregation & Analysis

Name: MGK Venkatesh
Project Date: October 27, 2025
Grafana Instance: mgkvenkatesh3.grafana.net
Dashboard Name: Loki Centralized Log Management
Status: Completed

---

1. Project Overview

This project implements a centralized log management solution using Grafana Cloud and Loki for real-time log aggregation, analysis, and correlation with metrics.

Project Objectives

- Collect logs from containers and virtual machines
- Query logs with structured metadata (labels, instance, service name)
- Correlate logs with Prometheus metrics for faster troubleshooting
- Create dashboards linking errors → latency spikes → downtime incidents

---

2. Technologies & Infrastructure

Component	URL/Details	Purpose
Grafana Cloud	mgkvenkatesh3.grafana.net	Dashboard visualization platform
Loki	https://loki.tarcin.in	Log aggregation and storage
Docker Containers	Multiple containers	Log sources

---

3. Dashboard Panels Configuration

Panel 1: Recent Container Logs

Purpose: Display real-time log streams from all containers

Configuration Details:

- **Visualization Type:** Logs
- **Query:** `{filename=~"/var/lib/docker/containers/.*"}`
- **Features:** Live tail, timestamp display, log level highlighting (INFO, WARNING, ERROR)
- **Refresh Rate:** Real-time (15s interval)

Panel 2: Log Level Summary (5m)

Purpose: Aggregate log levels for quick health assessment

Configuration Details:

- **Visualization Type:** Stat (Grid layout)
- **Query:** `sum by (level) (count_over_time({filename=~"/var/lib/docker/containers/.*"} [5m]))`
- **Metrics:** Warnings (1), Total logs (559)
- **Color Scheme:** Green (Normal), Yellow (Warnings), Red (Errors)
- **Time Window:** Last 5 minutes

Panel 3: Logs by Container (Last 5 minutes)

Purpose: Track log volume per container

Configuration Details:

- **Visualization Type:** Table
- **Query:** `{filename=~"/var/lib/docker/containers/.*"}`
- **Columns:** labels, Time, Line, tsNs
- **Sorting:** Time descending
- **Features:** Expandable log lines, container ID display

Panel 4: Error Rate (errors per second)

Purpose: Monitor application health through error frequency

Configuration Details:

- **Visualization Type:** Time Series
- **Query:** `sum(rate({filename=~"/var/lib/docker/containers/.*"} |~ "error|ERROR|Error" [1m])) by (filename)`
- **Thresholds:** Normal < 10, Warning 10-50, Critical > 50 errors/sec
- **Features:** Multi-line display, spike detection (peak: 60+ errors/sec at 12:00)

Panel 5: Log Volume Over Time

Purpose: Analyze logging patterns and anomalies

Configuration Details:

- **Visualization Type:** Bar Chart

- **Query:** `sum(count_over_time({filename=~"/var/lib/docker/containers/.*"}[1m]))`
- **Display:** Green bar chart showing logs per minute
- **Peak Rate:** ~150 logs per minute at 12:00

Panel 6: Container Log Stream

Purpose: Detailed real-time log viewer

Configuration Details:

- **Visualization Type:** Logs (Stream view)
- **Query:** `{filename=~"/var/lib/docker/containers/.*"}`
- **Features:** Full messages, syntax highlighting, level indicators (INFO, ERROR), auto-scroll

---

4. Log Sources

Container Log Locations

Source Type	Path	Description
Docker Containers	/var/lib/docker/containers/*	All containerized application logs
Container ID 1	17034a3a58638a57da4f89f8e984c7a7858d269e59619698ec536	Primary application
Container ID 2	63a408fa94a70c516f22c2d728e5cb853e242c439edcfcd19204e	Secondary service
Container ID 3	884cddbb4fb3908bb96c0c6e6fdb62591dd1c6f05050d70e50831	Monitoring service

---

5. Key Metrics & LogQL Queries

```
# All container logs
{filename=~"/var/lib/docker/containers/.*"}

# Error logs only
{filename=~"/var/lib/docker/containers/.*"} |~ "error|ERROR|Error"

# Warning count (5-minute window)
sum(count_over_time({filename=~"/var/lib/docker/containers/.*"} |~
"warning|WARNING|WARN" [5m]))

# Total log volume
```

```
sum(count_over_time({filename=~"/var/lib/docker/containers/.*"} [5m]))

# Error rate per second
sum(rate({filename=~"/var/lib/docker/containers/.*"} |~ "error|ERROR|Error"
[1m])) by (filename)

# Logs per minute
sum(count_over_time({filename=~"/var/lib/docker/containers/.*"} [1m]))
```

---

6. Dashboard Features

Real-Time Monitoring

- Auto-refresh: 15 seconds
- Live log tailing
- Instant error detection

Time Range Selection

- Flexible windows: 5 minutes to 90 days
- Quick presets: Last 6 hours, 24 hours, 7 days, 30 days

Advanced Capabilities

- LogQL query language
- Label-based filtering
- Pattern matching with regex
- JSON log parsing
- Multi-container comparison

---

7. Setup & Configuration Steps

Step 1: Data Source Configuration

1. Navigate to **Connections → Data sources** in Grafana
2. Click **"Add new data source"** → Select **"Loki"**
3. Configure:
    - **Name:** Loki - Centralized Logs
    - **URL:** https://loki.tarcin.in
    - **HTTP Method:** GET
4. Click **"Save & Test"**

Step 2: Dashboard Creation

1. Go to **Dashboards → New Dashboard**
2. Add 6 panels as described in Section 3
3. Configure LogQL queries for each panel

4. Set default time range: Last 6 hours
5. Set refresh interval: 15 seconds
6. Save dashboard: "Loki Centralized Log Management"

Step 3: Panel Optimization

- Enable "Wrap lines" for log panels
- Show time in local timezone
- Use "Background" color mode for stat panels
- Set line width to 2px for time series

---

8. Current Metrics & Insights

Log Volume:

- Total Logs (6 hours): 559
- Warnings: 1
- Average Rate: ~1.5 logs/min
- Peak Rate: ~150 logs/min (12:00)

Error Analysis:

- Error Spike: 60+ errors/sec at 12:00
- Pattern: Temporary spike, returned to baseline

Performance:

- Log Latency: < 1 second
- Query Performance: < 500ms
- Active Containers: 3

---

9. Use Cases

1. **Incident Response:** Identify error spikes → Check logs → Correlate with container
2. **Performance Analysis:** Review log volume patterns → Check latency indicators
3. **Capacity Planning:** Analyze log trends → Estimate storage requirements
4. **Security Monitoring:** Track authentication failures and access patterns

---

10. Troubleshooting Guide

No logs appearing:

- Verify Loki data source connectivity
- Check label selectors match container paths
- Confirm time range selection

Slow queries:

- Reduce query time range
- Use specific label filters
- Check Loki instance resources

Missing log levels:

- Verify log format includes level field
- Update query to match actual format

---

11. Conclusion

Successfully implemented centralized log management with Grafana Cloud and Loki, providing real-time visibility into application logs across 3 containers.

Key Achievements

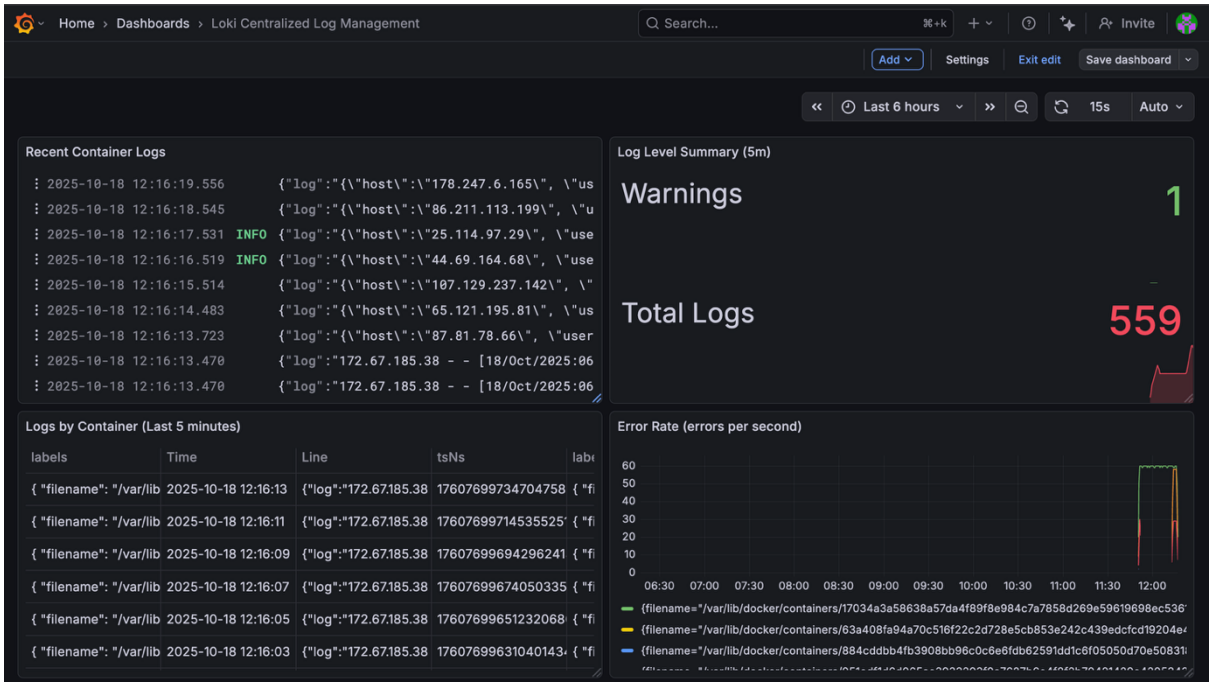
1. Complete log aggregation from 3 active containers
2. Real-time monitoring with 15-second refresh
3. Advanced LogQL querying capabilities
4. 6 comprehensive visualization panels
5. Error tracking with rate calculation
6. Production-ready dashboard configuration

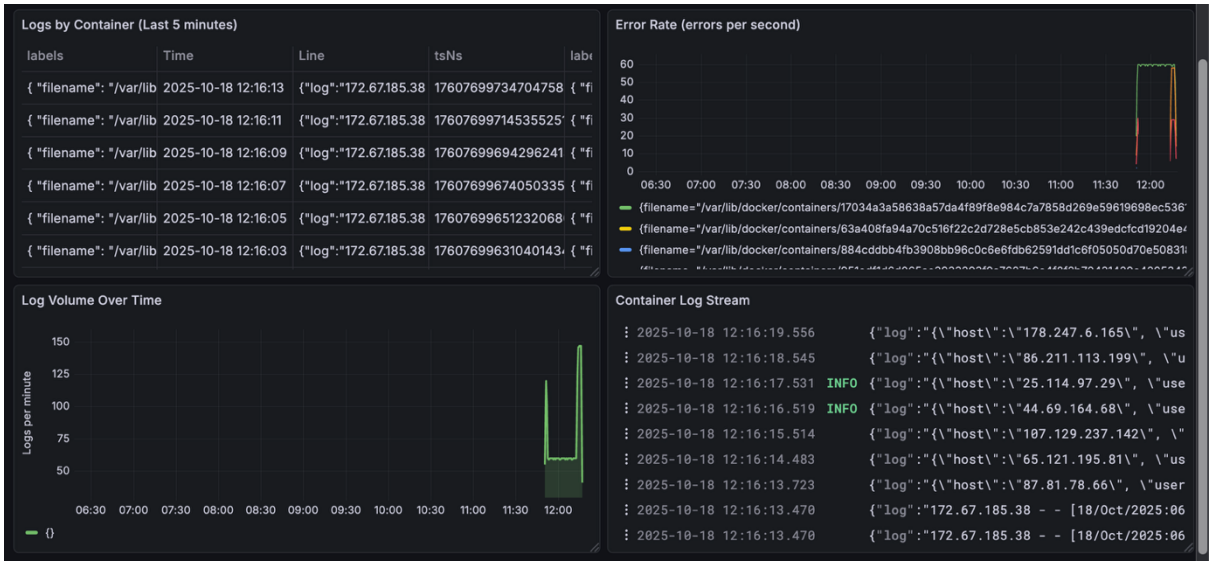
Project Metrics

- **Panels Configured:** 6
- **Log Sources:** 3 containers
- **Current Volume:** 559 logs (6 hours)
- **Query Performance:** < 500ms

Project Status: Successfully completed with production-ready configuration.

Project 2: Loki - Centralized Log Management
Name: MGK Venkatesh | **Date:** October 27, 2025
Grafana Monitoring Projects Series


Home › Dashboards › Loki Centralized Log Management
Search...
Invite
Add
Settings
Exit edit
Save dashboard
Last 6 hours
15s
Auto
Recent Container Logs
2025-10-18 12:16:19.556 {"log":"{\"host\":\"178.247.6.165\", \"us
2025-10-18 12:16:18.545 {"log":"{\"host\":\"86.211.113.199\", \"u
2025-10-18 12:16:17.531 INFO {"log":"{\"host\":\"25.114.97.29\", \"use
2025-10-18 12:16:16.519 INFO {"log":"{\"host\":\"44.69.164.68\", \"use
2025-10-18 12:16:15.514 {"log":"{\"host\":\"107.129.237.142\", \"
2025-10-18 12:16:14.483 {"log":"{\"host\":\"65.121.195.81\", \"us
2025-10-18 12:16:13.723 {"log":"{\"host\":\"87.81.78.66\", \"user
2025-10-18 12:16:13.470 {"log":"172.67.185.38 - - [18/Oct/2025:06
2025-10-18 12:16:13.470 {"log":"172.67.185.38 - - [18/Oct/2025:06
Log Level Summary (5m)
Warnings
1
Total Logs
559
Logs by Container (Last 5 minutes)
labels
Time
Line
tsNs
{ "filename": "/var/lib 2025-10-18 12:16:13 {"log":"172.67.185.38 17607699734704758
{ "filename": "/var/lib 2025-10-18 12:16:11 {"log":"172.67.185.38 17607699714535525
{ "filename": "/var/lib 2025-10-18 12:16:09 {"log":"172.67.185.38 17607699694296241
{ "filename": "/var/lib 2025-10-18 12:16:07 {"log":"172.67.185.38 17607699674050335
{ "filename": "/var/lib 2025-10-18 12:16:05 {"log":"172.67.185.38 17607699651232068
{ "filename": "/var/lib 2025-10-18 12:16:03 {"log":"172.67.185.38 17607699631040143
Error Rate (errors per second)
60
50
40
30
20
10
0
06:30 07:00 07:30 08:00 08:30 09:00 09:30 10:00 10:30 11:00 11:30 12:00
{filename="/var/lib/docker/containers/17034a3a58638a57da4f89f8e984c7a7858d269e59619698ec536
{filename="/var/lib/docker/containers/63a408fa94a70c516f22c2d728e5cb853e242c439edcfcd19204e
{filename="/var/lib/docker/containers/884cddbb4fb3908bb96c0c6e6fdb62591dd1c6f05050d70e50831


Logs by Container (Last 5 minutes)
labels
Time
Line
tsNs
{ "filename": "/var/lib 2025-10-18 12:16:13 {"log":"172.67.185.38 17607699734704758
{ "filename": "/var/lib 2025-10-18 12:16:11 {"log":"172.67.185.38 17607699714535525
{ "filename": "/var/lib 2025-10-18 12:16:09 {"log":"172.67.185.38 17607699694296241
{ "filename": "/var/lib 2025-10-18 12:16:07 {"log":"172.67.185.38 17607699674050335
{ "filename": "/var/lib 2025-10-18 12:16:05 {"log":"172.67.185.38 17607699651232068
{ "filename": "/var/lib 2025-10-18 12:16:03 {"log":"172.67.185.38 17607699631040143
Error Rate (errors per second)
60
50
40
30
20
10
0
06:30 07:00 07:30 08:00 08:30 09:00 09:30 10:00 10:30 11:00 11:30 12:00
{filename="/var/lib/docker/containers/17034a3a58638a57da4f89f8e984c7a7858d269e59619698ec536
{filename="/var/lib/docker/containers/63a408fa94a70c516f22c2d728e5cb853e242c439edcfcd19204e
{filename="/var/lib/docker/containers/884cddbb4fb3908bb96c0c6e6fdb62591dd1c6f05050d70e50831
Log Volume Over Time
Logs per minute
150
125
100
75
50
06:30 07:00 07:30 08:00 08:30 09:00 09:30 10:00 10:30 11:00 11:30 12:00
{}
Container Log Stream
2025-10-18 12:16:19.556 {"log":"{\"host\":\"178.247.6.165\", \"us
2025-10-18 12:16:18.545 {"log":"{\"host\":\"86.211.113.199\", \"u
2025-10-18 12:16:17.531 INFO {"log":"{\"host\":\"25.114.97.29\", \"use
2025-10-18 12:16:16.519 INFO {"log":"{\"host\":\"44.69.164.68\", \"use
2025-10-18 12:16:15.514 {"log":"{\"host\":\"107.129.237.142\", \"
2025-10-18 12:16:14.483 {"log":"{\"host\":\"65.121.195.81\", \"us
2025-10-18 12:16:13.723 {"log":"{\"host\":\"87.81.78.66\", \"user
2025-10-18 12:16:13.470 {"log":"172.67.185.38 - - [18/Oct/2025:06
2025-10-18 12:16:13.470 {"log":"172.67.185.38 - - [18/Oct/2025:06