

Index

Kubernetes Distribution	2
Necessity of Log Rotation in kubernetes :.....	2
1.0 Log Rotation of Health Check Logs :.....	5
1.2 Log Rotation of Applications Logs:.....	6
1.2.1 : Phrase from official Doc. :.....	6
Testing :.....	7
Observation Result:.....	9

Kubernetes Distribution I'am using : Microk8s

Necessity of Log Rotation in kubernetes :

1. Disk Space Management

- Prevents Log Files from Consuming Excessive Disk Space:
 - Continuous growth of log files without rotation can lead to significant consumption of disk space.
 - Ensures that nodes have sufficient storage available for other critical operations and data.
- Affects Pods:
 - If disk space runs out due to unbounded log growth, it can lead to pod eviction or prevent new pods from being scheduled.
 - Ensures that pods running on the node are not adversely affected by storage issues, maintaining the stability and availability of workloads.

2. Performance Optimization

- Improves File System Performance:
 - Smaller, rotated log files are easier and faster to read and write, reducing the performance impact on the node's filesystem.
 - Prevents slowdown of node operations due to handling of large log files.
- Affects Pods:
 - Improved node performance translates to better performance and responsiveness for the pods running on that node.

3. Improved Log Management

- Facilitates Easier Log Analysis and Troubleshooting:
 - Breaking down logs into manageable chunks simplifies log analysis and troubleshooting.
 - Enables better indexing and searching of log data.
- Affects Pods:
 - Easier and quicker access to logs helps in faster resolution of issues affecting pods, improving overall system reliability.

4. System Stability

- Avoids System Instability or Crashes:
 - Prevents system instability or crashes due to lack of disk space caused by unbounded log growth.
 - Maintains continuous operation of the cluster by ensuring healthy disk utilization.
- Affects Pods:
 - Stability of the node ensures that pods continue to run without interruption, avoiding downtime and disruptions in services.

5. Compliance and Auditing

- Meets Regulatory and Compliance Requirements:
 - Helps in retaining a manageable and orderly log history, meeting regulatory and compliance needs.
 - Facilitates auditing by maintaining a structured log retention policy.
- Affects Pods:
 - Ensuring compliance can be crucial for workloads that have regulatory requirements, affecting the legality and viability of running certain applications.

6. Resource Efficiency

- Optimizes Resource Usage:
 - Controls the size and number of log files, which is crucial in resource-constrained environments.
 - Ensures log rotation processes are efficient, minimizing overhead on the node.
- Affects Pods:
 - Efficient resource usage on nodes helps in better allocation of resources to pods, improving their performance and efficiency.

7. Enhanced Log Retrieval

- Simplifies Retrieval of Recent Logs:
 - Keeps the latest log files readily accessible, simplifying the process of log retrieval.
 - Ensures tools like kubectl logs can efficiently access the most relevant log data.
- Affects Pods:
 - Quick access to recent logs helps in monitoring and debugging issues with pods in a timely manner.

8. Preventing Log Corruption

- Reduces Risk of Log File Corruption:
 - Regular rotation reduces the risk of log file corruption that can occur with very large files.
 - Ensures the integrity and reliability of log data.
- Affects Pods:
 - Reliable log data is crucial for diagnosing and resolving issues with pods, ensuring their smooth operation.

9. Facilitating Backup and Archiving

- Easier to Backup and Archive Logs:
 - Smaller, rotated log files are easier to backup and archive compared to a single large file.
 - Allows for more granular control over log retention policies.
- Affects Pods:
 - Ensures that critical log data related to pod operations is safely archived and available for future reference or audits.

1.0 Log Rotation of Health Check Logs :

Little Explanation about this : In our project , we are writing health check logs from pod to hostpath.

Currently, all health check logs are written in the `/var/log/probe` hostPath directory.

Set up log rotation as follows:

File Name : `/etc/logrotate.d/k8s_probe_logs`

```
/var/log/probe/*/*.log {
    daily
    missingok
    rotate 30
    compress
    delaycompress
    notifempty
    create 0640 root root
    postrotate
        # Remove log files older than 90 days
        find /var/log/probe -type f -mtime +90 -delete
        # Remove directories (if empty after file deletion)
        find /var/log/probe -type d -empty -delete
    endscript
}
```

1.2 Log Rotation of Applications Logs:

By Default K8s written logs in this directory `/var/log/pods`

For my ref : <https://github.com/kubernetes/kubernetes/blob/master/cluster/gce/gci/configure-helper.sh>

Official doc.:

<https://kubernetes.io/docs/concepts/cluster-administration/logging/#:~:text=Log%20rotation&text=You%20can%20configure%20two%20kubelet,allowed%20for%20each%20container%20respectively.>

1.2.1 : Phrase from official Doc. :

Kubelet's Role in Log Rotation

- The kubelet is responsible for rotating container logs and managing the logging directory structure.
- The kubelet sends log rotation information to the container runtime (using CRI), and the runtime writes the container logs to the given location.

Configuring Log Rotation

- Two kubelet configuration settings can be used to configure log rotation & can be configure using `kubelet` configuration file :
 - `containerLogMaxSize` (default 10Mi): sets the maximum size for each log file
 - `containerLogMaxFiles` (default 5): sets the maximum number of files allowed for each container

Tuning Log Rotation

- Two additional kubelet configuration settings can be used to tune log rotation & can be configure using [kubelet configuration file](#) :
 - `containerLogMaxWorkers`: sets the number of concurrent log rotations that can be performed
 - `containerLogMonitorInterval`: sets the interval at which the logs are monitored and rotated as required

Note on kubectl logs

- Only the contents of the latest log file are available through `kubectl logs`.
- For example, if a Pod writes 40 MiB of logs and the kubelet rotates logs after 10 MiB, running `kubectl logs` returns at most 10MiB of data.

Testing :

Task: Create a Docker image that continuously writes logs, and then we will check whether the kubelet will rotate the logs or not.

Step 1:

Dockerfile:

```
# Use a base image with Python
FROM python:3.9-slim

# Add a script to generate logs
COPY generate_logs.py /generate_logs.py

# Set the command to run the script
CMD ["python", "/generate_logs.py"]
```

generate_logs.py :

```
import time
import logging

# Configure logging
logging.basicConfig(level=logging.INFO, format='%(asctime)s %(message)s')

# Define the log message
log_message = "This is a log message to fill up the log file. " * 10

# Generate logs indefinitely
while True:
    logging.info(log_message)
    time.sleep(0.01) # Sleep to prevent the script from running too fast
```

- Compile docker image and upload

pod.yaml :

```
apiVersion: v1
kind: Pod
metadata:
  name: log-generator
spec:
  restartPolicy: Always # Ensure the container restarts
  containers:
  - name: log-generator
    image: vishalk17/log-generator:v3
```

- Import that image in pod.yaml and configure pod.yaml in kubernetes

Step 2 : Check the default behavior of log rotation in microk8s (Note: I'm using microk8s distribution of kubernetes)

```
root@vishalk17:~# cd /var/log/pods/default_log-generator_809780b8-cb12-4557-a6a5-cfeb11f96162/log-generator
root@vishalk17:~# watch ls -ltha
total 23M
-rw-r----- 1 root root 7.1M Jun  6 00:29 1.log
drwxr-xr-x 2 root root 4.0K Jun  6 00:27 .
-rw-r--r-- 1 root root 245K Jun  6 00:27 1.log.20240606-002407.gz
-rw-r----- 1 root root 11M Jun  6 00:27 1.log.20240606-002728
-rw-r--r-- 1 root root 245K Jun  6 00:24 1.log.20240606-002047.gz
-rw-r--r-- 1 root root 245K Jun  6 00:20 1.log.20240606-001727.gz
-rw-r----- 1 root root 4.4M Jun  6 00:10 0.log
drwxr-xr-x 3 root root 4.0K Jun  6 00:09 ..
```

Observation Result:

- Log rotation works only for active pods.
- Log rotation keeps a maximum of 5 log files; beyond that, it deletes the oldest one.
- The maximum log file size is 11 MB.
- If a pod is deleted, its logs and the log directory of the pod are also deleted.