



CSCI 5410

Serverless Data Processing (Summer 2023)

MASTER OF APPLIED COMPUTER
SCIENCE

Assignment-2 (Part A): Summary of Research Paper

Name: **Jainil Sevalia** | Banner Id : **B00925445** | Email: jn498899@dal.ca

Summary:

The author of the research paper, Nitin Naik, presents an evaluation of the high availability and fault tolerance capabilities of Docker Swarm, a distributed system for container management [1]. The paper demonstrates how Docker Swarm effectively manages the failover of the Swarm leader, ensuring the normal operation of the cluster even in the event of leader failures [1]. The system achieves this through additional managers and a quorum of managers that can promote a new leader.

In the experiments conducted by the author, the behaviour of Docker Swarm during leader failures is evaluated [2]. When the leader on one Docker machine fails, the cluster promotes another manager as the new leader, maintaining normal operation. However, when the leader on a different Docker machine fails and only one manager remains active, the cluster cannot operate normally without the required quorum of managers [1]. By reinstating a manager, the cluster regains the quorum and successfully promotes a new leader, restoring normal operation.

Nitin Naik highlights that Docker Swarm, utilizing the Raft Consensus Algorithm for manager coordination, offers fault tolerance for a maximum of "f" manager failures when the cluster incorporates $2f+1$ managers [1]. The loss of the quorum restricts certain actions, such as adding or removing swarm nodes and starting or updating tasks, but existing tasks on worker nodes continue to run. Docker Swarm also ensures worker node availability, with node failures only affecting individual worker nodes rather than the entire cluster [2].

The paper further discusses the automatic scalability, load balancing, and maintainability features of Docker Swarm [1]. The author conducted a test scenario where services were created on the Swarm cluster's nodes and easily scaled based on system requirements. Docker Swarm effectively balanced the load across the cluster and allocated replicas to nodes with the best available resources [2]. The system also demonstrated maintainability by automatically switching instances to operational nodes when nodes failed.

Additionally, the paper mentions recent research studies comparing the scalability performance of Docker Swarm and Google Kubernetes in large cluster environments [1], [3]. These studies involved running 30,000 containers across 1,000 nodes in a cluster. The comparison focused on container startup time and system responsiveness under load [1]. Docker Swarm exhibited an average five-fold faster container startup time and seven-fold faster system responsiveness compared to Google Kubernetes.

In conclusion, Research paper highlights the high availability, fault tolerance, automatic scalability, load balancing, and maintainability capabilities of Docker Swarm. The experiments and studies conducted by the author demonstrate the effectiveness of Docker Swarm in managing distributed systems and its superior performance in certain scenarios when compared to Google Kubernetes [1],[2],[3].

References:

- [1] N. Naik, "Performance Evaluation of Distributed Systems in Multiple Clouds using Docker Swarm," 2021 IEEE International Systems Conference (SysCon), Vancouver, BC, Canada, 2021, pp. 1-6, doi: 10.1109/SysCon48628.2021.9447123
- [2] "Docker Swarm," Docker Documentation. [Online]. Available: <https://docs.docker.com/engine/swarm/> [Accessed Jul. 2, 2023].
- [3] "GKE Overview; Google Kubernetes Engine (GKE); google cloud," Google. [Online]. Available: <https://cloud.google.com/kubernetes-engine/docs/concepts/kubernetes-engine-overview> [Accessed Jul. 2, 2023].