

**ANALISIS PERBANDINGAN PERFORMA *CONTAINER* PADA  
KUBERNETES *SERVICE* DI *PLATFORM* GOOGLE CLOUD PLATFORM  
(GCP), AMAZON WEB SERVICES (AWS), DAN MICROSOFT AZURE**

**SKRIPSI**

Diajukan Untuk Memenuhi Sebagian Syarat Untuk Memperoleh  
Gelar Sarjana Teknik Program Studi Sistem Telekomunikasi  
Universitas Pendidikan Indonesia



Oleh

Palda Puspita Dianti

1904211

**PROGRAM STUDI SISTEM TELEKOMUNIKASI  
KAMPUS UPI DI PURWAKARTA  
UNIVERSITAS PENDIDIKAN INDONESIA  
2023**

**ANALISIS PERBANDINGAN PERFORMA *CONTAINER* PADA  
KUBERNETES *SERVICE* DI *PLATFORM* GOOGLE CLOUD PLATFORM  
(GCP), AMAZON WEB SERVICES (AWS), DAN MICROSOFT AZURE**

Oleh  
Palda Puspita Dianti

Sebuah skripsi yang diajukan untuk memenuhi salah satu syarat memperoleh gelar  
Sarjana Teknik Program Studi Sistem Telekomunikasi

Universitas Pendidikan Indonesia

© Palda Puspita Dinati 2023  
Universitas Pendidikan Indonesia  
Agustus 2023

Hak Cipta dilindungi undang-undang.

Skripsi ini tidak boleh diperbanyak seluruhnya atau sebagian,  
dengan dicetak ulang, difoto kopi, atau cara lainnya tanpa ijin dari penulis.

**LEMBAR PENGESAHAN**

PALDA PUSPITA DIANTI

**ANALISIS PERBANDINGAN PERFORMA *CONTAINER* PADA  
KUBERNETES *SERVICE* DI *PLATFORM* GOOGLE CLOUD PLATFORM  
(GCP), AMAZON WEB SERVICES (AWS), DAN MICROSOFT AZURE**

Disetujui dan Disahkan Oleh:

Pembimbing I:



**Galura Muhammad Suranegara, S.Pd., M.T.**

**NIP. 920190219920111101**

Pembimbing II



**Hafiyvan Putra Pratama, S.ST., M.T.**

**NIP. 920190219921224101**

Mengetahui,

Ketua Program Studi Sistem Telekomunikasi  
UPI Kampus Purwakarta



**Galura Muhammad Suranegara, S.Pd., M.T.**

**NIP. 920190219920111101**

**ANALISIS PERBANDINGAN PERFORMA *CONTAINER* PADA KUBERNETES  
SERVICE DI PLATFORM GOOGLE CLOUD PLATFORM (GCP), AMAZON  
WEB SERVICES (AWS), DAN MICROSOFT AZURE**

**PALDA PUSPITA DIANTI**

**NIM. 1904211**

**ABSTRAK**

*Containerisasi* merupakan teknologi yang sedang berkembang pesat dan memiliki kemungkinan akan terus menjadi tren di beberapa tahun ke depan. Akibatnya, sistem *Container Orchestration* menjadi semakin penting. Dan dari berbagai *Container Orchestration* yang ada, Kubernetes lah yang menjadi standar *de facto*. Hal ini dikarenakan *stability*, *maturity*, dan fungsionalitasnya yang komprehensif. Semua penyedia *Cloud* utama pada saat ini menawarkan solusi Kubernetes terkelola *Cloud-Native* untuk membebaskan pengguna dari tekanan karena harus merancang dan memelihara infrastruktur Kubernetes yang rumit sambil tetap memanfaatkan fungsinya. Pada tahun 2022, 96% dari Sysdig's global *customer* yang menggunakan layanan *container* memilih Kubernetes sebagai *Container Orchestration* nya. Dan dalam laporan awal CNCF pada tahun 2022 sebanyak 79% responden menggunakan layanan terkelola seperti GKE, EKS, dan AKS. Dalam penelitian ini, tes dilakukan didalam *cluster* Kubernetes berisi *container* dengan 2 *node* pada *instance machine* dengan 4vcpu yang di-*deploy* dengan *nginx:1.23* pada 3 *Cloud Service Provider* yaitu GCP, AWS, dan Azure. Pengujian performa dilakukan secara *real time* menggunakan Prometheus, Grafana, dan Python. Hasil dari pengujian menempatkan GCP sebagai platform terbaik dikarenakan unggul dalam berbagai aspek performa seperti *execution time*, efisiensi jaringan dan *price performance*.

**Kata kunci:** *Cloud Computing*, *Container*, Kubernetes, GCP, AWS, Azure, GKE, EKS, AKS, *Performance test*

**COMPARISON ANALYSIS OF CONTAINER PERFORMANCE ON KUBERNETES  
SERVICE AT GOOGLE CLOUD PLATFORM (GCP), AMAZON WEB SERVICES  
(AWS), AND MICROSOFT AZURE**

**PALDA PUSPITA DIANTI**

**NIM. 1904211**

***ABSTRACT***

*Containerization is a continuously developing technology that will most likely remain popular in the next years. As a result, the Container Orchestration system has grown in significance. Kubernetes is the de facto standard among the numerous existing Container Orchestration. This is because of its overall stability, maturity, and functionality. Today, all major Cloud providers provide Cloud-Native managed Kubernetes solutions, relieving users of the burden of designing and maintaining complicated Kubernetes infrastructure while still accessing its capability. By 2022, 96% of Sysdig's global container customers used Kubernetes as their container orchestration. In the initial CNCF report in 2022, 79% of respondents used managed services such as GKE, EKS, and AKS. In this study, tests were performed in a Kubernetes cluster with 2 nodes on machine instances with 4 vCPU deployed with nginx:1.23 on three Cloud Service providers, namely GCP, AWS, and Azure. Prometheus, Grafana, and Python are used for real-time performance Testing. According to the findings of the testing, GCP is the best platform since it excels in a variety of performance metrics such as execution time, network efficiency, and price performance.*

**Keyword:** *Cloud Computing, Container, Kubernetes, GCP, AWS, Azure, GKE, EKS, AKS, Performance test*

## DAFTAR ISI

LEMBAR PENGESAHAN .....	i
LEMBAR PERNYATAAN .....	ii
KATA PENGANTAR.....	iii
UCAPAN TERIMAKASIH .....	iv
ABSTRAK .....	vi
<i>ABSTRACT</i> .....	vii
DAFTAR ISI .....	viii
DAFTAR GAMBAR .....	xii
DAFTAR TABEL.....	xv
DAFTAR LAMPIRAN .....	xvi
DAFTAR ISTILAH .....	xvii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Dan Batasan Masalah.....	2
1.2.1 Rumusan Masalah.....	2
1.2.2 Batasan Masalah .....	3
1.3 Tujuan Penelitian.....	3
1.4 Manfaat Penelitian .....	3
1.4.1 Manfaat Teoritis .....	3
1.4.2 Manfaat Praktis .....	3
1.4.2.1 Bagi Penulis .....	3
1.4.2.2 Bagi Pengembangan Ilmu .....	4
1.4.2.3 Bagi Perusahaan.....	4
1.5 Sistematika Penulisan Skripsi .....	4

1.5.1	Bagian Awal Skripsi.....	4
1.5.2	Bagian Utama Skripsi.....	4
1.5.3	Bagian Akhir Skripsi.....	5
BAB II TINJAUAN PUSTAKA.....		6
2.1	<i>Cloud Computing</i> .....	6
2.1.1	Infrastructure as a Service (IaaS).....	6
2.1.2	<i>Platform as a Service (PaaS)</i> .....	7
2.1.3	<i>Software as a Service (SaaS)</i> .....	7
2.2	<i>Container</i> .....	8
2.2.1	Definisi <i>Container</i> .....	8
2.2.1.1	<i>Deployment Tradisional</i> .....	8
2.2.1.2	<i>Deployment Container</i> .....	9
2.2.2	<i>Container Orchestration</i> .....	10
2.2.2.1	<i>Container Orchestration Tools</i> .....	10
2.2.2.2	<i>Container Orchestration Work</i> .....	11
2.2.2.3	<i>Monitoring</i> .....	11
2.3	Docker.....	11
2.4	Kubernetes.....	12
2.4.1	<i>Hosted</i> .....	12
2.4.2	Self-Hosted.....	13
2.4.3	Arsitektur Kubernetes.....	13
2.4.3.1	<i>Master Node</i> .....	14
2.4.3.2	<i>Working Node</i> .....	15
2.5	<i>Cloud Service Provider (CSPs)</i> .....	16
2.5.1	Google Cloud Platform (GCP).....	16
2.5.1.1	Google Kubernetes Engine (GKE).....	16

2.5.2	Amazon Web Service (AWS) .....	17
2.5.2.1	Amazon Elastic Kubernetes Service (EKS).....	17
2.5.3	Microsoft Azure .....	17
2.5.3.1	Azure Kubernetes Service (AKS).....	17
2.6	Penelitian Relevan.....	17
BAB III METODE PENELITIAN.....		20
3.1	Jenis Penelitian.....	20
3.2	Desain Penelitian.....	22
3.3	Teknik Pengumpulan Data .....	23
3.3.1	Studi Pustaka.....	23
3.3.2	Observasi .....	23
3.4	Teknik Analisis Data .....	24
3.4.1	Analisis Kebutuhan.....	24
3.4.2	Desain Sistem .....	25
3.4.2.1	Arsitektur GKE .....	25
3.4.2.2	Arsitektur EKS.....	26
3.4.2.3	Arsitektur AKS.....	27
3.4.3	Implementasi Desain.....	28
3.4.4	<i>Integration dan Testing</i> Sistem .....	28
3.4.4.1	<i>Benchmarking</i> .....	28
3.4.4.2	<i>Observability</i> .....	29
3.4.5	<i>Operation dan Maintenance</i> .....	30
3.5	Waktu dan Tempat Penelitian.....	30
3.5.1	Waktu Penelitian .....	30
3.5.2	Tempat Penelitian .....	30
BAB IV HASIL DAN PEMBAHASAN.....		31



4.1	<i>Deploy Aplikasi</i> .....	31
4.1.1	Cara Membuat <i>Cluster</i> di GKE .....	31
4.1.2	Cara Membuat Cluster di EKS .....	34
4.1.3	Cara Membuat <i>Cluster</i> di AKS.....	38
4.2	<i>Benchmarking</i> .....	42
4.2.1	<i>CrashLoopBackOff</i> .....	42
4.3	<i>Observability</i> .....	43
4.3.1	<i>Deploy Prometheus dan Ambil Data</i> di GKE .....	43
4.3.2	<i>Deploy Prometheus dan Grafana</i> di EKS.....	45
4.3.3	<i>Deploy Prometheus dan Ambil data</i> di AKS.....	50
4.4	Hasil .....	53
4.4.1	<i>Deployment</i> .....	53
4.4.2	CPU.....	54
4.4.3	<i>Memory</i> .....	59
4.4.4	<i>Network</i> .....	61
4.4.5	<i>Analisis Cost</i> .....	64
BAB V PENUTUP.....		67
5.1	Kesimpulan .....	67
5.2	Implikasi.....	68
5.3	Rekomendasi.....	68
DAFTAR PUSTAKA .....		69
LAMPIRAN.....		74
RIWAYAT HIDUP .....		95

## DAFTAR PUSTAKA

- Arango, C., Dernat, R., & Sanabria, J. (2017). *Performance Evaluation of Container-based Virtualization for High Performance Computing Environments* (arXiv:1709.10140). arXiv. <http://arxiv.org/abs/1709.10140>
- Birje, M. N., & Bulla, C. (2020). Commercial and Open Source Cloud Monitoring Tools: A Review. Dalam S. C. Satapathy, K. S. Raju, K. Shyamala, D. R. Krishna, & M. N. Favorskaya (Ed.), *Advances in Decision Sciences, Image Processing, Security and Computer Vision* (Vol. 3, hlm. 480–490). Springer International Publishing. [https://doi.org/10.1007/978-3-030-24322-7\\_59](https://doi.org/10.1007/978-3-030-24322-7_59)
- Borén, F. W. (2021). *Case study: Performance evaluation of Kind*. <https://www.diva-portal.org/smash/get/diva2:1600690/FULLTEXT01.pdf>
- Deepika, T., & Rao, A. N. (2014). Active resource provision in cloud computing through virtualization. *2014 IEEE International Conference on Computational Intelligence and Computing Research*, 1–4. <https://doi.org/10.1109/ICCIC.2014.7238373>
- Ginting, J. G. A., Ikhwan, S., & Ammar, M. N. (2021). Analisis Performansi High Availability Web Server Pada Cluster GKE (Google Kubernetes Engine) Menggunakan Infrastruktur Google Cloud Platform. *InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan*, 5, 9.
- Isaias, P., & Issa, T. (2015). *High Level Models and Methodologies for Information Systems*. Springer New York. <https://doi.org/10.1007/978-1-4614-9254-2>
- Jawarneh, I. M. A., Bellavista, P., Bosi, F., Foschini, L., Martuscelli, G., Montanari, R., & Palopoli, A. (2019). Container Orchestration Engines: A Thorough Functional and Performance Comparison. *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 1–6. <https://doi.org/10.1109/ICC.2019.8762053>

- Kamal, M. A., Raza, H. W., Alam, M. M., & Su'ud, M. M. (2020). Highlight the Features of AWS, GCP and Microsoft Azure that Have an Impact when Choosing a Cloud Service Provider. *International Journal of Recent Technology and Engineering (IJRTE)*, 8(5), 4124–4132.  
<https://doi.org/10.35940/ijrte.D8573.018520>
- Kopytov, A. (2023, Maret). SysBench. *GitHub*.  
<https://github.com/akopytov/sysbench>
- Kubernetes. (2022a, November 4). API Overview [Kubernetes.io]. *Kubernetes.Io*.  
<https://kubernetes.io/docs/reference/using-api/>
- Kubernetes. (2022b, Desember 5). Kubernetes Scheduler [Kubernetes.io]. *Kubernetes.Io*. <https://kubernetes.io/docs/concepts/scheduling-eviction/kube-scheduler/>
- Kubernetes. (2023a, April 12). Kube-controller-manager [Kubernetes.io]. *Kubernetes.Io*. <https://kubernetes.io/docs/reference/command-line-tools-reference/kube-controller-manager/>
- Kubernetes. (2023b, April 23). Kubelet [Kubernetes.io]. *kubernetes.io*.  
<https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/>
- Kumar, M. S., & S, Dr. S. (2022). A Comprehensive Comparison Study on Container Orchestration Frameworks. *International Journal of Research in Engineering and Science (IJRES)*, 10(5), 72–77.
- Liu, X., Shen, W., Liu, B., Li, Q., Deng, R., & Ding, X. (2019). Research on Large Screen Visualization Based on Docker. *Journal of Physics: Conference Series*, 1169, 012052. <https://doi.org/10.1088/1742-6596/1169/1/012052>
- Massey, V., & Satao, K. (2012). Comparing various SDLC models and the new proposed model on the basis of available methodology. *International*

*Journal of Advanced Research in Computer Science and Software Engineering*, 2(4), 170–177.

- Nugroho, M. A. (2018). Analisis Cluster Container Pada Kubernetes Dengan Infrastruktur Google Cloud Platform. *JIPi (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 3(2). <https://doi.org/10.29100/jipi.v3i2.651>
- Pan, Y., Chen, I., Brasileiro, F., Jayaputera, G., & Sinnott, R. (2019). A Performance Comparison of Cloud-Based Container Orchestration Tools. *2019 IEEE International Conference on Big Knowledge (ICBK)*, 191–198. <https://doi.org/10.1109/ICBK.2019.00033>
- Pereira Ferreira, A., & Sinnott, R. (2019). A Performance Evaluation of Containers Running on Managed Kubernetes Services. *2019 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, 199–208. <https://doi.org/10.1109/CloudCom.2019.00038>
- Poulton, N. (2017). *The Kubernetes Book*. 13–28.
- Preeth E N, Mulerickal, Fr. J. P., Paul, B., & Sastri, Y. (2015). Evaluation of Docker containers based on hardware utilization. *2015 International Conference on Control Communication & Computing India (ICCC)*, 697–700. <https://doi.org/10.1109/ICCC.2015.7432984>
- Redhat. (2022, Mei 10). Understanding Linux Containers. <https://www.redhat.com/>.  
<https://www.redhat.com/en/topics/containers/what-is-container-orchestration>
- RedHat. (2022, Mei 13). *What is Grafana?*  
<https://www.redhat.com/en/topics/data-services/what-is-grafana>
- Reinartz, F. (2023). Prometheus. *GitHub*.  
<https://github.com/prometheus/prometheus>

- Rice, L. (2023, Juli). Kube-bench. *GitHub*. <https://github.com/aquasecurity/kube-bench>
- Saraswat, M., & Tripathi, R. C. (2020). Cloud Computing: Comparison and Analysis of Cloud Service Providers-AWs, Microsoft and Google. *2020 9th International Conference System Modeling and Advancement in Research Trends (SMART)*, 281–285. <https://doi.org/10.1109/SMART50582.2020.9337100>
- Saraswathi, A. T., Kalaashri, Y. R. A., & Padmavathi, S. (2015). Dynamic Resource Allocation Scheme in Cloud Computing. *Procedia Computer Science*, 47, 30–36. <https://doi.org/10.1016/j.procs.2015.03.180>
- Shah, J., & Dubaria, D. (2019). Building Modern Clouds: Using Docker, Kubernetes & Google Cloud Platform. *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, 0184–0189. <https://doi.org/10.1109/CCWC.2019.8666479>
- Shukur, H., Zeebaree, S., Zebari, R., Zeebaree, D., Ahmed, O., & Salih, A. (2020). Cloud Computing Virtualization of Resources Allocation for Distributed Systems. *Journal of Applied Science and Technology Trends*, 1(3), 98–105. <https://doi.org/10.38094/jastt1331>
- Vailshery, L. S. (2021). *Container Orchestration Platform Do You Primarily Use on 2020*.
- Velayudhan, N. (2021, September 1). What are container runtimes? *Opensource.Com*. <https://opensource.com/article/21/9/container-runtimes>
- Whiston, T. (2019, Maret 28). Classic Deployment vs. Container Deployment. *Nine Cloud Navigators*. <https://www.nine.ch/en/blog/classic-deployment-vs-container-deployment-new#:~:text=The%20traditional%20style%20of%20deployment,stability%20are%20guaranteed%20without%20interruptions.>

- Zaki, K., & Saad, H. (2018). Adoption of Cloud Human Resource Information System in Egyptian Hotels: An Experimental Design Research. *International Journal of Heritage, Tourism and Hospitality*, 12(1), 233–245. <https://doi.org/10.21608/ijhth.2018.31514>
- Zhang, M. (2023). *Top 10 Cloud Service Providers Globally in 2023*. <https://dgtlinfra.com/top-10-cloud-service-providers-2022/>
- Zhang, Y., Yin, G., Wang, T., Yu, Y., & Wang, H. (2018). An Insight Into the Impact of Dockerfile Evolutionary Trajectories on Quality and Latency. *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, 138–143. <https://doi.org/10.1109/COMPSAC.2018.00026>