UNIVERSITAS
GADJAH MADA
Implementasi dan Evaluasi Kinerja autoscaling pada Klaster Kubernetes Berbasis K3s Menggunakan Horizontal Pod Autoscaler dengan Beban Kerja ETL Pipeline
Fadhlan Anfasal Hajji, Dr. Ir. Ronald Adrian, S.T., M.Eng., IPM.
Universitas Gadjah Mada, 2026 | Diunduh dari http://etd.repository.ugm.ac.id/

INTISARI

IMPLEMENTASI DAN EVALUASI KINERJA *AUTOSCALING* PADA KLASTER KUBERNETES BERBASIS K3S MENGGUNAKAN HORIZONTAL POD AUTOSCALER DENGAN BEBAN KERJA *ETL PIPELINE*

Fadhlan Anfasal Hajji

22/497009/SV/21071

Adopsi arsitektur berbasis *container* serta orkestrasi Kubernetes menuntut mekanisme *autoscaling* yang andal untuk menjaga efisiensi serta responsivitas sistem. Horizontal Pod Autoscaler (HPA) merupakan mekanisme *autoscaling* bawaan Kubernetes yang menyesuaikan jumlah replika *pod* berdasarkan metrik utilisasi sumber daya. Penelitian ini mengimplementasikan dan mengevaluasi kinerja *autoscaling* berbasis HPA pada klaster K3s yang di-*deploy* secara *self-hosted* di atas tiga Virtual Private Server (VPS). *Pipeline ETL* berbasis Python digunakan sebagai pemicu beban yang bersifat *stateless* dengan volume data yang divariasikan ke dalam empat ukuran, yaitu 1, 10, 100, dan 500 MB, bersumber dari ulasan aplikasi transportasi umum Jakarta. Evaluasi mencakup latensi *scale-out*, lama *delay* deteksi respon data HPA, waktu respons *pipeline ETL*, serta penggunaan sumber daya CPU dan memori. Hasil pengujian menunjukkan HPA baru melakukan *autoscaling* pada beban 100 MB ke atas, dengan latensi *scale-out* yang stabil pada kisaran 6–7 detik tanpa dipengaruhi ukuran data. Waktu respons *pipeline ETL* justru meningkat tajam seiring beban, dari 5,76 detik pada 1 MB hingga 1.652,88 detik pada 500 MB, sedangkan lama *delay* deteksi HPA dan penggunaan sumber daya turut bertambah hingga sistem mencapai tiga *pod* pada beban terbesar. Temuan ini mengindikasikan faktor pembatas utama kinerja terletak pada durasi pemrosesan ETL, bukan pada mekanisme *autoscaling* HPA yang berlangsung cepat dan konsisten. Penelitian ini memberikan kontribusi empiris pada karakteristik *autoscaling* HPA di lingkungan Kubernetes *self-hosted* berskala kecil sekaligus menjadi acuan praktis dalam pengaturan parameter HPA.

Kata kunci: *Autoscaling,* Horizontal Pod Autoscaler*, K3s, Pipeline ETL, Load Testing*

UNIVERSITAS
GADJAH MADA
Implementasi dan Evaluasi Kinerja autoscaling pada Klaster Kubernetes Berbasis K3s Menggunakan Horizontal Pod Autoscaler dengan Beban Kerja ETL Pipeline
Fadhlan Anfasal Hajji, Dr. Ir. Ronald Adrian, S.T., M.Eng., IPM.
Universitas Gadjah Mada, 2026 | Diunduh dari http://etd.repository.ugm.ac.id/

ABSTRACT

IMPLEMENTATION AND PERFORMANCE EVALUATION OF AUTOSCALING ON A K3S-BASED KUBERNETES CLUSTER USING HORIZONTAL POD AUTOSCALER WITH ETL PIPELINE WORKLOAD

Fadhlan Anfasal Hajji
22/497009/SV/21071

The adoption of container-based architectures and Kubernetes orchestration demands a reliable auto-scaling mechanism to maintain system efficiency and responsiveness. Horizontal Pod Autoscaler (HPA) is a built-in Kubernetes auto-scaling mechanism that adjusts the number of pod replicas based on resource utilization metrics. This study implements and evaluates the performance of HPA-based auto-scaling on a K3s cluster deployed in a self-hosted configuration across three virtual private servers (VPS). The load is generated by a stateless Python-based ETL pipeline whose data volume is programmatically varied into four sizes, namely 1, 10, 100, and 500 MB, sourced from user reviews of Jakarta public transportation applications. The evaluation covers scale-out latency, HPA detection delay, ETL pipeline response time, and CPU and memory resource usage. The results show that HPA only performs auto-scaling at loads of 100 MB and above, with scale-out latency remaining stable in the range of 6–7 seconds regardless of data size. In contrast, ETL pipeline response time increases sharply with load, from 5.76 seconds at 1 MB to 1,652.88 seconds at 500 MB, while HPA detection delay and resource usage also grow until the system reaches three pods at the largest load. These findings indicate that the primary performance-limiting factor lies in the ETL processing duration rather than in the HPA auto-scaling mechanism, which runs quickly and consistently. This study provides empirical insight into the characteristics of HPA-based auto-scaling in small-scale self-hosted Kubernetes environments and serves as a practical reference for configuring HPA parameters.

Keywords: *Autoscaling, Horizontal Pod Autoscaler, K3s, Pipeline ETL, Load Testing*