LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 1, No. 5, Agustus 2023
ISSN 2985-4172 (media online)
Hal 1084-1090

IMPLEMENTASI WEBSERVER BERBASIS DOCKER DAN LINUX

Antonio Hati Dame[1], Achmad Udin Zailani[2]

[1,2] Fakultas Ilmu Komputer, Program Studi Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia
E-mail: [1]hatidame@gmail.com , [2]dosen00270@unpam.ac.id

Abstrak- Teknologi Docker countainer dan jenis Bahasa pemrograman kini kiat berpengaruh dalam mengembangkan aplikasi, dengan menggunkan Docker sebagai teknologi container kita dapat menghemat resource jika di bandingkan dengan teknologi virtualisasi seperti Virtualbox, VMWare. Linux adalah nama yang diberikan kepada kumpulan system operasi mirip unix sebagai kernelnya, linux merupakan perangkat lunak bebas dan sumber terbuka terbesar di dunia, seperti perangkat lunak bebas dan sumber terbuka lainnya pada umumnya, kode sumber linux dapat dimodifikasi, dan didistribusikan kembali secara bebas oleh siapa saja dan biaya operasional yang rendah, dan kompatiblitas yang tinggi dibandingkan system unix yang tak bebas.

Kata Kunci: Implementasi Web Server, Docker, Linux

Abstract- *Countainer technology and type of programming language are now influential tips in developing applications, by using Docker as a technology container we can increase resources when compared to virtualization technologies such as Virtualbox, VMWare. Linux is the name given to a collection of unix-like operating systems as the kernel, linux is the largest free and open source software in the world, like other free and open source software in general, linux source code can be transferred, and returned freely by anyone and low operating costs, and high compatibility with non-free Unix systems.*

Keywords: *Web Server Implementation, Docker, Linux*

1. PENDAHULUAN

Dalam membangun program, pengembang biasanya menjalankan *virtualisasi* pada server sehingga proses pembuatan program dapat berjalan pada berbagai platform maupun konfigurasi *hardware*. Masalah yang dihadapi dengan *virtualisasi* adalah perlunya menyiapkan satu sistem operasi secara utuh, termasuk berbagai aplikasi yang dibawa sistem tersebut. Bisa dibayangkan dengan banyaknya *virtualisasi* yang berjalan di sebuah server akan memberatkan sistem tersebut.

Teknik docker *continer* adalah teknik dimana mengisolasi suatu sistem sehingga tidak mengganggu sistem yang lainnya, Salah satu teknik *virtualisasi* berbasis linux dan *container*, *Virtualisasi* berbasis linux dan *container* adalah teknik yang tidak menerapkan *Hypervisior* yang hanya mengisolasikan proses tanpa mengisolasi perangkat keras, Kernel dan *operating system* sehingga dapat mengurangi *overhead* pada *hardware* juga performanya lebih baik dari *virtualisas*i mesin salah satu *virtualisasi* berbasis linux dan *container* adalah Docker.

Dengan diterapkannya sistem *virtualisasi* server berbasis docker *container*, diharapkan dapat meningkatkan kinerja sebuah server dan memudahkan proses *deployment* (penyebaran) aplikasi web beserta *software* pendukung seperti *webserver*, *database server*, dll ke server. Maka permasalahan yang ada diantaranya adalah bagaimana mengimplementasikan docker untuk pengelolaan *webserver* Jauh daripada itu, akan tercapai efektifitas dan efisiensi kerja mesin karena sumber daya mesin akan dgunakan secara maksimal untuk menjalankan layanan yang ada berbasis virtual.

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 1, No. 5, Agustus 2023
ISSN 2985-4172 (media online)
Hal 1084-1090

2. METODE

2.1. Metode Penelitian

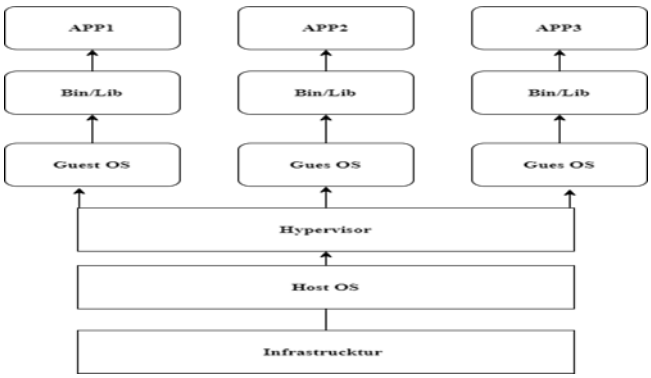
Metode penelitian yang akan digunakan dalam penelitian ini meliputi :

a. Studi literatur
   Mencari bahan-bahan yang tersedia berdasarkan referensi dari jurnal, buku, artikel ilmiah, sumber dari internet maupun penelitian terdahulu yang ada kaitannya dengan Implementasi Webserver Berbasis Linux Dan Docker.
b. Konsultasi dan diskusi
   Penulis juga berkonsultasi dan berdiskusi dengan orang yang ahli dalam sistem linux
c. Perancangan
   Perancangan yang dimaksudkan untuk mendapatkan hasil yang baik.
d. Implementasi
   Melakukan implementasi sistem dan skenario yang telah di buat dan mengoptimalisasi agar sesuai dengan spesifikasi yang diinginkan.
e. Pengujian
   Melakukan pengujian pada webserver dengan mengunakan Node Exporter,Pormetheus Monitoring dan grafana sebagai alat pengujian webserver.
f. Pembuatan Laporan
   Setiap proses yang dilakukan akan ditulis dalam bentuk laporan atau gambar.

2.2. Metode Sistem Berjalan Saat Ini

Saat ini sistem yang berjalan di webserver menggunkan *virtual machine,* masih menggunakan cara yang tradisional yaitu :

a. Server adalah sistem computer yang memiliki layanan khusus berupa penyimpanan data. Server akan menyimpan beragam jenis dokumen dan menyediakan informasi untuk penguna.
b. *Hypervisor* adalah salah satu teknik *virtualisasi* yang bertugas untuk mebagi-bagikan sumber daya dan mengalokasikan ke berberapa sistem operasi/*virtual machine* yang berbeda.
c. *Guest Os* adalah Sebuah sistem operasi yang berjalan di bawah kendali *hypervisor.*
d. *Libraries* adalah kumpulan kode yang biasanya terkumpul dalam sebuah *namespace/module/package* (tergantung menggunakannya dibahasa pemrograman apa) yang dapat di *import* ke program lain.
e. Aplikasi adalah perangkat lunak yang beroperasi secara independen dari fungsionalnalitas teknisi sistem operasi, aplikasi menyediakan fungsi yang hanya tersedia jika menginstal aplikasi.



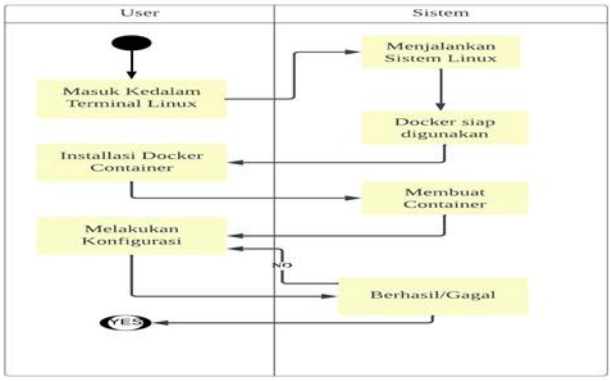
Gambar 1. Sistem Berjalan Saat Ini

Dengan menggunakan konsep *Virtual machine* pada sistem *webserver* akan menghabiskan *resource* yang digunakan, karena konsep *virtual machine* ini yaitu berjalan diatas *OS* dan membuat *OS*

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 1, No. 5, Agustus 2023
ISSN 2985-4172 (media online)
Hal 1084-1090

terbaru untuk *virtual machie* yang dimana ini akan membebankan kinerja dari perangkat komputer itu sendiri.

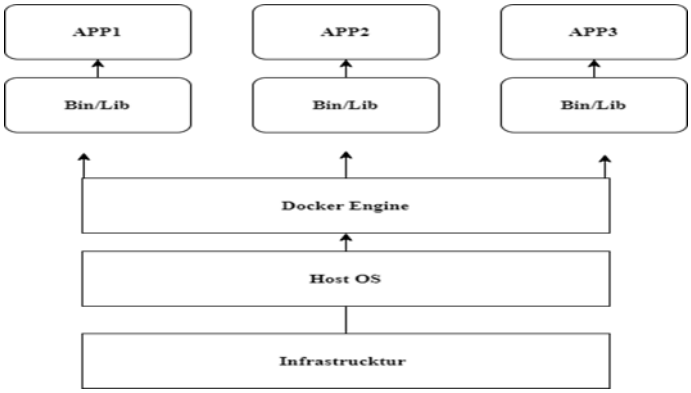
2.3. Metode Sistem Perancangan Usulan



Gambar 2. *Activity Diagram* Konfigurasi Webserver

Peneliti mengusulkan untuk merancang sebuah webserver yang akan dibuat serta penjelasan alurnya sebagai berikut:

a. Menyiapkan beberapa perangkat Lunak seperti, Vmware workstion, Mobaxtrem, Sistem operasi linux yang akan digunakan sebagai webserver, dan sistem windows yang digunakan sebagai client untuk mengakses sebuah webserve, website php dan database mysql.
b. Melakukan installasi dan Konfigurasi pada Docker Container.
c. Melakukan installasi webserver Nginx dan konfigurasi file nginx dan php agar bisa saling terkoneksi.
d. Mengakses di windows browser hasil konfigurasi Nginx.



Gambar 3. Sistem Perancangan Usulan

Dengan melihat sistem yang sedang berjalan saat ini membuat *webserver* menggunakan *virtual machine* sebagai *heypervisor* diatas *host* os akan membebankan sebuah server dan akan menghabiskan *resource* yang digunakan dan sistem usulan saat ini yang saya terapakan tidak menggunakan *hypervisor* di atas *host os* tapi menggunakan teknologi docker *container* sebagai membuat *webserver.*

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 1, No. 5, Agustus 2023
ISSN 2985-4172 (media online)
Hal 1084-1090

3. ANALISA DAN PEMBAHASAN

3.1. Pegujian Fungsional

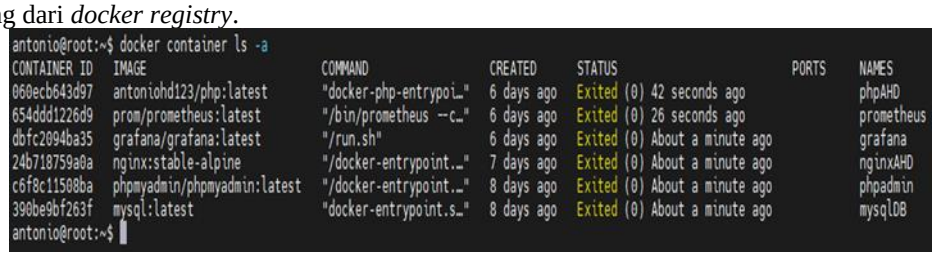
Setelah Perancangan, installasi dan konfigurasi telah dibuat, kemudian hasil dari perancangan, installasi dan konfigurasi tersebut diimplementasikan sebagai berikut. Images digunakan untuk menjalakan sebuah *container*, maka dari itu pada tahapan ini penulis menginstall *images* dari docker *registry* dan *images* dari php akan *dibuild* dengan menggunakan *Dockerfile* untuk menjalankan *container* php dan menjalankan semua *container* yang sudah dibuat oleh penulis.

Gambar 4. *Docker Images*

Pada gambar diatas adalah hasil dari menginstall *images* dari docker *regsirty*

Gambar 5. *Images* Hasil Membangun (*build*)

Gambar diatas adalah hasil membangun (*build*) *images* php dengan Dockerfile tanpa menginstall langsung dari *docker registry*.



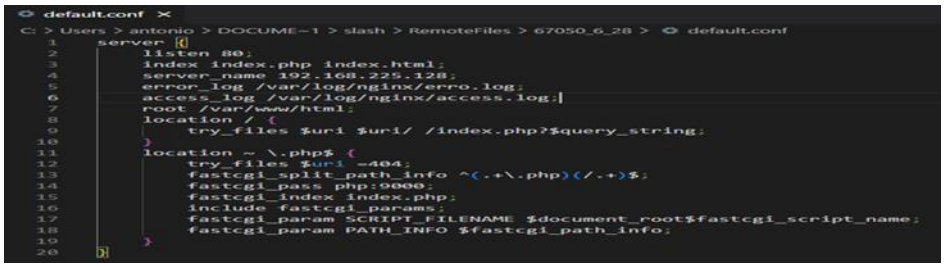
Gambar 6. *Docker Container*

Gambar diatas adalah Docker *container* yang sudah dibuat (*create*) dari docker *images* dengan nama dan *port* dll.

Gambar 7. Webserver Nginx

Gambar diatas adalah *webserver* nginx yang sudah terkoneksi melalui komputer client dengan menggunakan Browser dikomputer *client* bahwa *container* yang dijalanakan berhasil menjalankan Nginx sebagai webserver.

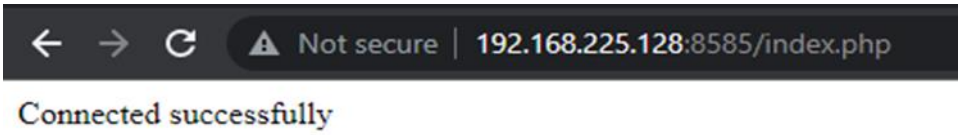
LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 1, No. 5, Agustus 2023
ISSN 2985-4172 (media online)
Hal 1084-1090



Gambar 8. Konfigurasi Nginx

Gambar diatas adalah konfigurasi nginx bertujuan agar nginx bisa membaca dan memproses file php.

Gambar 9. Tampilan File Index.php



Gambar 10. Tampilan Webserver Nginx

Gambar pertama adalah file index.php agar *webserver* nginx bisa membaca dan memproses file php dan saling terhubung webserver nginx aplikasi php dan mysql database. Gambar kedua adalah melihat hasil apakah webserver sudah berjalan dengan php melalui dikomputer client menggunakan *browser.*

3.2. Pengujian Webserver

Pada tahap ini, pengujian dilakukan untuk mengecek setiap fungsi yang ada pada webserver dengan monitoring maupun melakukan test mematikan dan menyalakan aplikasi telah berjalan sesuai dengan semestinya dan beberapa tahapan pengujian baik itu melalui host docker ataupun komputer client yang terkoneksi dengan host docker berikut adalah langkah dan hasil pengujian webserver, hasil dari pengujian yang dilakukan dengan interaksi antara client server dan server adalah sebagai berikut:

Tabel 1. Pengujian Webserver

No	Kebutuhan Fungsional	Hasil
1	Penulis melakukan remote akses ke *server* dengan login menggunakan SSH	ok
2	Penulismenambahkan *i*mage baru dari sebuah aplikasi ke server dari docker registry	ok
3	Penulis membuat container baru dari sebuah image yang telah ditambahkan dari docker registry	ok

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 1, No. 5, Agustus 2023
ISSN 2985-4172 (media online)
Hal 1084-1090

4	Penulis menjalankan container yang telah dibuat	ok
5	Penulis menghentikan container yang telah dibuat	ok
6	Penulis menghapus container yang telah dibuat	ok
7	Penulis berhasil mengkases docker metrics	ok
8	Penulis berhasil mengkonfigurasikan docker metrics dan node exporter ke prometheus	ok
9	Penulis berhasil membuat akun di dashboard grafana	ok
10	Penulis berhasil menghubungkan prometheus ke dashboard grafana	ok

3.3. Hasil Penggunaan CPU

Penggunaan CPU yang diukur adalah penggunan cpu paling tinggi pada setiap pengujian menggunakan banyak nya jumlah *user* yang berbeda pada masing masing *container* yang berjalan. Data yang diambil berasal dari statistik web yang sedang berjalan ketika proses pengujian penggunan cpu, pada penelitian ini dapat dilihat pada tabel dan gambar dibawah ini

Tabel 2. Hasil Penggunaan Cpu

User Request	Nama *Container*	
	Web 1	Web 2
200	2.54 %	4.19 %
400	4.99 %	6.49 %
600	7.54 %	8.58 %
800	10.62 %	10.84 %
1000	12.35 %	13.89 %

Gambar 11. Grafik Hasil Penggunaan Memori

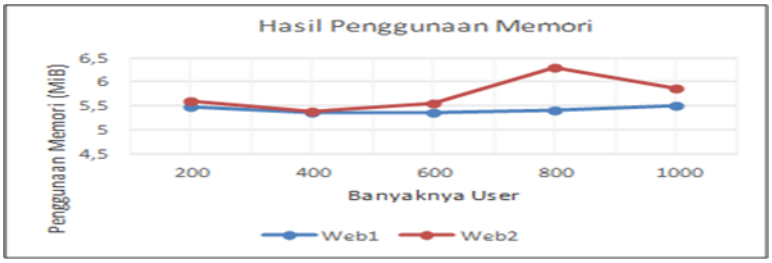
3.4. Hasil Penggunaan Memori

Penggunaan memori yang diukur adalah penggunaan memori paling tinggi pada setiap pengujian menggunakan variasi banyaknya jumlah *user* yang berbeda pada masing-masing *container* yang berjalan. Data yang diambil berasal dari statistik web yang sedang berjalan ketika proses pengujian berlangsung. pada penelitian ini dapat dilihat pada tabel dan gambar.

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 1, No. 5, Agustus 2023
ISSN 2985-4172 (media online)
Hal 1084-1090

Tabel 3. Hasil Penggunaan Memori

User	Nama *Container*	
Request	Web 1	Web 2
200	5.492 MiB	5.598 MiB
400	5.367 MiB	5.383 MiB
600	5.367 MiB	5.559 MiB
800	5.398 MiB	6.297 MiB
1000	5.508 MiB	5.875 MiB



Gambar 12. Grafik Hasil Penggunaan Memori

4. KESIMPULAN

Berdasarkan hasil implementasi, webserver berbasis docker *container* dapat mengurangi *resource* pada server hardware maupun *software* dan mampu menyediakan *environment* yang stabil untuk dijalanakan di perangkat manapun dengan konsep tekonologi *containerized*, dapat mempermudah pekerjaan *developer* ketika mengembangkan aplikasi.

REFERENSI

Afandi, A. T. (2020). Network Monitoring Berbasis Docker Container. *Implementasi Network Monitoring Sistem Menggunakan Libernms Berbasis Dcoker Container*, vol. 13, no. 01, (2021)

Alauddin, M. F. (2017). Virtual Data Center . *Implementasi Virtual Data Center Mengunakan Linux Container Berbasis Docker Dan Sdn*, Vol. 6, No. 2 (2017), 2337-3520

Aziz, A. (2020). Webserver Nginx Dengan Lighttpd. *Analisa Perbandingan Kinerja Webserver Nginx Dengan Lighttpd Untuk Kebutuhan Manajemen Web*,

Bik, M. F. (2017). Pengelolan Banyak Aplikasi. *Implementasi Docker Untuk Penglolaan Banyak Aplikasi*, Vol 7 no 2 Tahun 2017, 46-50

Fs Ariadi, C. I. (2020). Docker Container Sebagai Teknologi. *Penerapan Docker Container Sebagai Teknologi Ramah Skalabilitas Dibanding Teknik Virtualisasi Untuk Membangun Website Dubuntu 18.04 LTS*, vol 8 no2 (2020): vol 8 no 2 Desember 2020

Harianto, A. D. (2020). Webserver Apache Dan Nginx. *Performasi Webserver Apache Dan Ngiinx Pada Aplikasi Penjualan Online*, *Vol 9 No 3 – Juni– 2020*

Pulu, O. S. (n.d.). Docker Sebagai Sstem Pada Aplikasi. *Implementasi Docker Sebagai sistem Pada Aplikasi*,

Rama Aria Megantara1, F. A. (2021). Implementasi Docker. *Implementasi Docker Dalam Membantu Virtual Lab Pada Universitas*, Science And Engineering National Seminar 6 (SENS 6)- Semarang, 16 Desember 2021

Saddiq, H. (2021). Load Balancing Webserver Menggunakan Container. *Implementasi Dan Pengukuran Performasi Load Balacing Web Server Menggunakan Container*, 20-35.

Sardi, E. S. (2017). Virtualisasi Container Dengan Docker. *Implementasi Teknik Virtualisasi Container Dengan Docker Untuk Pengelolaan Aplikasi Web Dinas Komunikasi Dan Informatika Kota Payakumbuh*,