

BAB III METODOLOGI PENELITIAN

3.1 Objek dan Subjek Penelitian

Objek penelitian ini berfokus pada perbandingan performa dari web *server* (*Nginx*) dengan subyek penelitian berupa *container Docker* dan *Podman*.

3.2 Alat dan Bahan Penelitian

Penelitian ini melibatkan perangkat lunak, perangkat keras, dan perangkat virtualisasi. Sebuah laptop dengan karakteristik berikut ini digunakan sebagai perangkat keras:

- a. Perangkat keras
 1. Intel® Core™ i3-1005G1 @1.20 GHz (4 CPUs)
 2. RAM 8 GB
- b. Perangkat virtualisasi

Tabel 3. 1 Spesifikasi Virtualisasi

<i>Container</i>	<i>Processor</i>	<i>RAM</i>	<i>Alamat IP</i>
<i>Docker</i>	2 CPUs	2048 MB	192.168.56.2
<i>Podman</i>	2 CPUs	2048 MB	192.168.56.3

- c. Perangkat lunak yang digunakan adalah sebagai berikut:

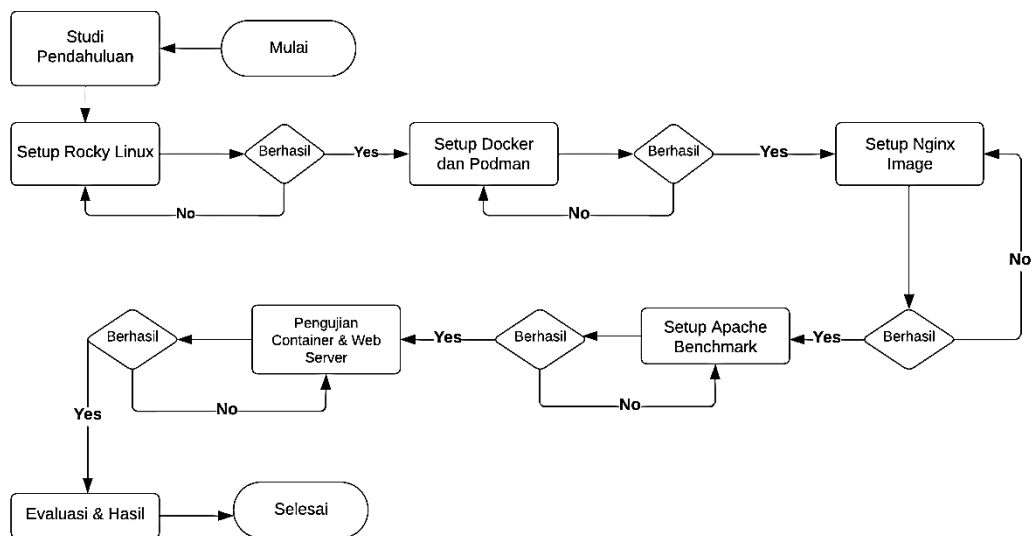
Tabel 3. 2 Perangkat Lunak

NO	<i>Software</i>	Versi	Fungsi
1	Oracle VM VirtualBox	7.0	Digunakan sebagai virtualisasi rocky linux.
2	OS Rocky Linux	8.6	Digunakan sebagai media instalasi <i>container</i> , web <i>server</i> dan alat pengujian.
3	Apache Benchmark	2.3	Digunakan sebagai pengujian dan penelitian.

NO	Software	Versi	Fungsi
4	Prometheus	2.46.0	Digunakan sebagai sistem <i>monitoring</i> dan disambungkan ke grafana.
5	Grafana	10.0.2	digunakan sebagai visualisasi dari <i>server monitoring</i> prometheus.
6	Microsoft Word	2019	Digunakan sebagai media penulisan.
7	Microsoft Excel	2019	Digunakan untuk mengolah data.
8	Chrome	114	Digunakan sebagai alat pembantu penelitian.

3.1 Diagram alir penelitian

Diagram alir penelitian digambarkan pada Gambar 3.1



Gambar 3. 1 Diagram Alir Penelitian

3.3.1 Studi Pendahuluan

Tahap awal ini dilakukan perumusan masalah berdasarkan studi penelitian yang ada. Studi penelitian berasal dari berbagai sumber valid seperti jurnal, buku dengan tahun terbit 2019 keatas yang berkaitan dengan permasalahan yang dihadapi.

3.3.2 Setup Rocky Linux

Pada proses ini dilakukan instalasi Rocky Linux pada *Virtual Machine* VirtualBox untuk menjalankan *Operating System* yang akan digunakan.

3.3.3 Setup Docker dan Podman

Tahap ini dilakukan instalasi dan konfigurasi *Container Docker* dan *Podman* pada Rocky Linux yang sudah di konfigurasi sebelumnya.

a. Docker

Gambar 3.2 merupakan konfigurasi awal yang dilakukan dalam proses instalasi *Docker*.

```
[root@herwin herwin]# dnf update
[root@herwin herwin]# sudo dnf install docker-ce docker-ce-cli containerd.io
```

Gambar 3. 2 Instalasi Docker

Gambar 3.3 merupakan verifikasi bahwa *Docker* berhasil diinstal.

```
[root@herwin herwin]# docker version
Client: Docker Engine - Community
Version: 23.0.1
API version: 1.42
Go version: go1.19.5
Git commit: a5ee5b1
Built: Thu Feb 9 19:49:07 2023
OS/Arch: linux/amd64
Context: default

Server: Docker Engine - Community
Engine:
Version: 23.0.1
API version: 1.42 (minimum version 1.12)
Go version: go1.19.5
Git commit: bc3805a
Built: Thu Feb 9 19:46:47 2023
OS/Arch: linux/amd64
Experimental: false
containerd:
Version: 1.6.19
GitCommit: 1e1ea6e986c6c86565bc33d52e34b81b3e2bc71f
runc:
Version: 1.1.4
GitCommit: v1.1.4-0-g5fd4c4d
docker-init:
Version: 0.19.0
GitCommit: de40ad0
```

Gambar 3. 3 Versi Docker

Gambar 3.4 merupakan *Docker images* yang berfungsi sebagai *history images* yang telah terunduh.

```
[root@herwin herwin]# docker images
REPOSITORY      TAG          IMAGE ID       CREATED        SIZE
nginx-server    latest      6ee91ea5fd1d  8 days ago    142MB
server-nginx    latest      6ee91ea5fd1d  8 days ago    142MB
[root@herwin herwin]#
```

Gambar 3. 4 Docker Images

Gambar 3.5 merupakan *Docker ps* yang berfungsi untuk melihat *images* yang sedang berjalan.

```
ER ID  IMAGE      COMMAND          CREATED        STATUS        PORTS
52722  server-nginx  "/docker-entrypoint..."  15 hours ago  Up 15 hours  0.0.0.0:80->80/tcp, :::80-
arvabhata
```

Gambar 3. 5 Docker Ps

b. Podman

Gambar 3.6 merupakan konfigurasi awal yang dilakukan dalam proses instalasi *Podman*.

```
[root@herwin herwin]# dnf update
[root@herwin herwin]# dnf install -y podman
```

Gambar 3. 6 Instalasi Podman

Gambar 3.7 merupakan verifikasi bahwa *Podman* berhasil diinstal.

```
[root@herwin herwin]# podman version
Client:      Podman Engine
Version:     4.2.0
API Version: 4.2.0
Go Version:  go1.18.9
Built:      Tue Feb 21 13:16:34 2023
OS/Arch:    linux/amd64
[root@herwin herwin]#
```

Gambar 3. 7 Versi Podman

Gambar 3.8 merupakan *Podman images* yang berfungsi untuk melihat *images* yang telah di unduh.

```
[root@herwin herwin]# podman images
REPOSITORY      TAG          IMAGE ID       CREATED        SIZE
localhost/server-nginx  latest      72fe1c717141  9 days ago    146 MB
docker.io/library/nginx  latest      ac232364af84  13 days ago   146 MB
[root@herwin herwin]#
```

Gambar 3. 8 Podman Images

Gambar 3.9 merupakan *Podman ps* yang berfungsi untuk melihat images yang sedang berjalan.

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
a12a82fa6dbf	localhost/server-nginx:Latest	nginx -g daemon o...	12 hours ago	Up 12 hours ago	0.0.0.0:80->80/tcp	hungry_chaplygin

Gambar 3. 9 Podman Ps

3.3.4 Setup Nginx

Pada tahap ini, dilakukan konfigurasi *DockerFile* untuk menjalankan *container Nginx*. Kemudian konfigurasi *Web Server Nginx* agar dapat berjalan dan berfungsi pada *Container Docker* maupun *Podman*.

Berikut merupakan konfigurasi dari file *Nginx.conf* dan *Dockerfile* yang dipakai.

```
worker_processes 1;
events {
    worker_connections 1024;
}
http {
    server {
        listen 80;
        server_name localhost;
        location / {
            root /usr/share/Nginx/html;
            index index.html;
        }
    }
}
```

```
FROM Nginx
COPY Nginx.conf /etc/Nginx/Nginx.conf
COPY index.html /usr/share/Nginx/html
EXPOSE 80
CMD ["Nginx", "-g", "daemon off;"]
```

3.3.5 Setup Apache Benchmark

Pada tahap ini dilakukan Instalasi *tools* Apache Benchmark yang nantinya digunakan untuk pengujian dan perbandingan.

Berikut merupakan *syntax* yang digunakan dalam proses instalasi :

```
#apt-get update
#yum install httpd-tools
```

Apt-get update digunakan untuk memperbarui daftar paket serta peningkatan versi yang diperlukan. *Yum install httpd-tools* digunakan sebagai *syntax* instalasi *tools* Apache Benchmark.

```
Benchmarking 192.168.56.3 (be patient)

Server Software:      nginx/1.23.3
Server Hostname:     192.168.56.3
Server Port:         80

Document Path:       /
Document Length:     517 bytes

Concurrency Level:   50
Time taken for tests: 1.077 seconds
Complete requests:   2500
Failed requests:     0
Total transferred:   1875000 bytes
HTML transferred:    1292500 bytes
Requests per second: 2322.26 [#/sec] (mean)
Time per request:    21.531 [ms] (mean)
Time per request:    0.431 [ms] (mean, across all concurrent requests)
Transfer rate:       1700.87 [Kbytes/sec] received

Connection Times (ms)
      min  mean[+/-sd] median  max
Connect:    1   10  4.3    9   25
Processing:  2   11  3.8   11   24
Waiting:    2   10  3.7   10   24
Total:      8   21  6.0   20   39

Percentage of the requests served within a certain time (ms)
 50%    20
 66%    23
 75%    24
 80%    26
 90%    29
 95%    32
 98%    35
 99%    38
100%    39 (longest request)
```

Gambar 3. 10 Hasil Pengujian

ab -n 2500 -c 50 <http://192.168.56.3/>

- Parameter *n* adalah jumlah koneksi yang dibuat ke *Server* tujuan, dengan contoh di atas berarti koneksi yang dibuat adalah 2500 koneksi.
- Parameter *c* adalah jumlah *request concurrent* (bersamaan) yang dibuat, dengan contoh di atas berarti jumlah *request* yang dibuat adalah 50 *request* dalam satu waktu.
- Parameter terakhir adalah *Address*. *Address* dapat berupa alamat IP atau halaman yang akan diproses oleh web *Server* benchmark.

3.3.6 Pengujian *Container* dan Web *Server*

Di tahap ini dilakukan pengujian pada web *server* dengan melihat *response time* dari *request* akses yang diminta oleh *user* dengan nilai *concurrent* 50, 100, 150, 250 dengan pengujian menggunakan *Docker* maupun *Podman*. Pengujian juga dilanjutkan dengan melihat penggunaan RAM dan CPU pada masing-masing *container*.

3.3.7 Evaluasi dan Hasil

Tahap evaluasi menjelaskan bahwa proses pengujian yang dilakukan dari awal hingga akhir akan dianalisis dengan melihat hasil perbandingan kinerja web *server* berdasarkan waktu respon, penggunaan RAM dan CPU. Kemudian menarik kesimpulan berdasarkan hasil analisis tersebut.