I. Testing

1. Round Robin

```
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking jarkom.site (be patient).....done


Server Software:        hcdn
Server Hostname:        jarkom.site
Server Port:            80

Document Path:          /
Document Length:        10066 bytes

Concurrency Level:      100
Time taken for tests:   1.087 seconds
Complete requests:      100
Failed requests:        0
Total transferred:      1037600 bytes
HTML transferred:       1006600 bytes
Requests per second:    91.97 [#/sec] (mean)
Time per request:       1087.294 [ms] (mean)
Time per request:       10.873 [ms] (mean, across all concurrent requests)
Transfer rate:          931.93 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:       21   44   7.9     43      66
Processing:    44  190 137.2    177     996
Waiting:       23  104  41.6     99     204
Total:         65  235 139.7    224    1043

Percentage of the requests served within a certain time (ms)
  50%    224
  66%    255
  75%    267
  80%    276
  90%    293
  95%    509
  98%    982
  99%   1043
 100%   1043 (longest request)
```

2. Least Connection

```
root@Revolte:/# ab -n 100 -c 100 http://jarkom.site/
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking jarkom.site (be patient).....done


Server Software:        hcdn
Server Hostname:        jarkom.site
Server Port:            80

Document Path:          /
Document Length:        10066 bytes

Concurrency Level:      100
Time taken for tests:   0.878 seconds
Complete requests:      100
Failed requests:        0
Total transferred:      1037600 bytes
HTML transferred:       1006600 bytes
Requests per second:    113.87 [#/sec] (mean)
Time per request:       878.184 [ms] (mean)
Time per request:       8.782 [ms] (mean, across all concurrent requests)
Transfer rate:          1153.84 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:       18    49  16.0     49      75
Processing:    50   225 129.4    216     755
Waiting:       29   152  60.9    176     271
Total:         69   274 136.3    284     830

Percentage of the requests served within a certain time (ms)
  50%    284
  66%    320
  75%    326
  80%    331
  90%    389
  95%    577
  98%    775
  99%    830
 100%    830 (longest request)
root@Revolte:/#
```

3. IP Hash

```
root@Revolte:/# ab -n 100 -c 100 http://jarkom.site/
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking jarkom.site (be patient).....done


Server Software:        hcdn
Server Hostname:        jarkom.site
Server Port:            80

Document Path:          /
Document Length:        10066 bytes

Concurrency Level:      100
Time taken for tests:   0.620 seconds
Complete requests:      100
Failed requests:        0
Total transferred:      1037600 bytes
HTML transferred:       1006600 bytes
Requests per second:    161.40 [#/sec] (mean)
Time per request:       619.584 [ms] (mean)
Time per request:       6.196 [ms] (mean, across all concurrent requests)
Transfer rate:          1635.42 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:       19    42   8.5     45      59
Processing:    84   185  75.8    180     528
Waiting:       23   109  43.5    106     217
Total:        116   227  79.4    226     577

Percentage of the requests served within a certain time (ms)
  50%    226
  66%    272
  75%    281
  80%    288
  90%    301
  95%    325
  98%    466
  99%    577
 100%    577 (longest request)
root@Revolte:/# 
```

4. Generic Hash

```
root@Revolte:/# ab -n 100 -c 100 http://jarkom.site/
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking jarkom.site (be patient).....done


Server Software:        hcdn
Server Hostname:        jarkom.site
Server Port:            80

Document Path:          /
Document Length:        10066 bytes

Concurrency Level:      100
Time taken for tests:   0.571 seconds
Complete requests:      100
Failed requests:        0
Total transferred:      1037600 bytes
HTML transferred:       1006600 bytes
Requests per second:    175.27 [#/sec] (mean)
Time per request:       570.564 [ms] (mean)
Time per request:       5.706 [ms] (mean, across all concurrent requests)
Transfer rate:          1775.93 [Kbytes/sec] received

Connection Times (ms)
             min  mean[+/-sd] median   max
Connect:       18   46  13.1     52     128
Processing:    41  192  85.6    191     500
Waiting:       20  104  44.0    100     231
Total:         59  238  89.6    241     533

Percentage of the requests served within a certain time (ms)
  50%    241
  66%    274
  75%    288
  80%    303
  90%    340
  95%    380
  98%    521
  99%    533
 100%    533 (longest request)
root@Revolte:/#
```
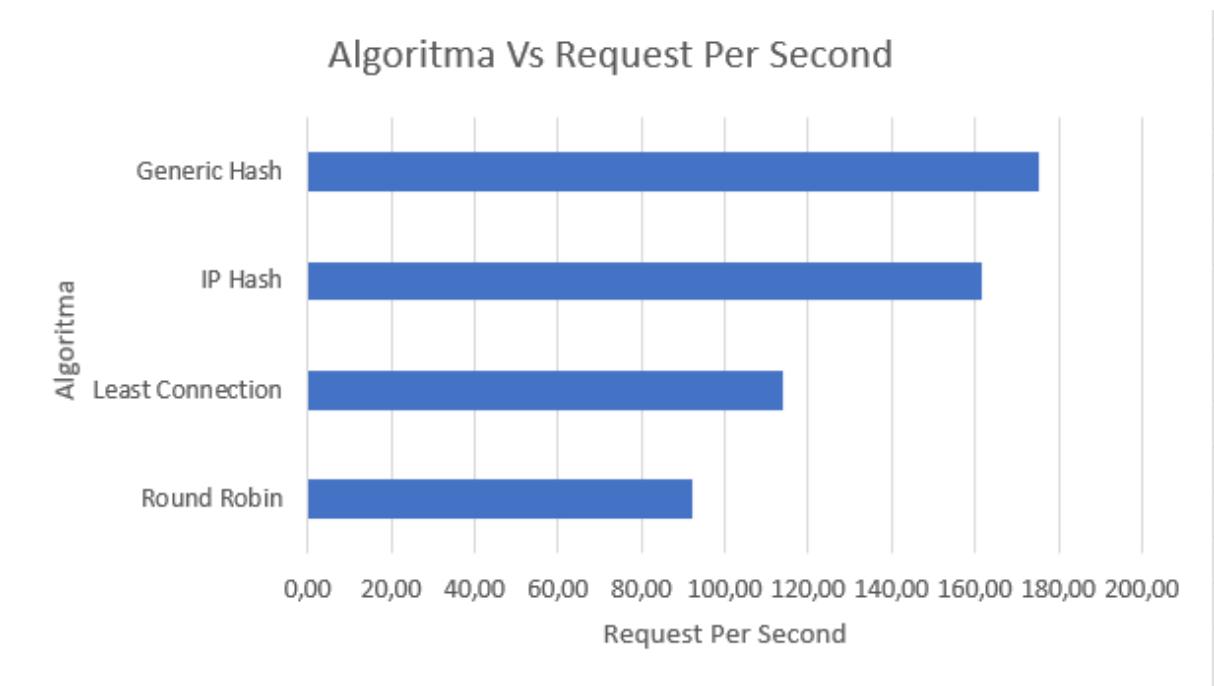
II. Grafik

Algoritma Vs Request Per Second

Request Per Second (X axis, 0,00 – 200,00)

Algoritma (Y axis):
- Generic Hash: ~175
- IP Hash: ~162
- Least Connection: ~115
- Round Robin: ~92

III. Analisis

1. Round Robin

Kelebihan:
- Sederhana dan mudah diimplementasikan.
- Setiap server mendapatkan jumlah permintaan yang hampir sama.

Kekurangan:
- Tidak mempertimbangkan beban aktual server, dapat menyebabkan ketidakseimbangan.

2. Least Connection
Kelebihan:
- Mendelegasikan permintaan ke server dengan koneksi terendah, sehingga dapat mendistribusikan beban secara lebih merata.
- Efektif untuk server yang memiliki kapasitas berbeda.

Kekurangan:
- Membutuhkan pemantauan koneksi yang konstan.

3. IP Hash

Kelebihan:

- Mendistribusikan permintaan berdasarkan alamat IP pengguna, memastikan konsistensi dalam distribusi beban.
- Berguna jika pengguna perlu dipetakan ke server tertentu.

Kekurangan:

- Tidak efektif jika alamat IP pengguna memiliki pola ketidakseimbangan.

4. Generic Hash

Kelebihan:

- Menggunakan fungsi hash untuk mendistribusikan permintaan, memberikan keacakan yang baik.
- Efektif untuk mendistribusikan beban secara merata pada server.

Kekurangan:

- Memerlukan implementasi fungsi hash yang baik.

Kesimpulan:

- Generic Hash menunjukkan kinerja RPS tertinggi di antara algoritma yang disajikan.
- Least Connection juga menunjukkan kinerja yang baik dengan distribusi beban yang lebih merata.
- Round Robin sederhana tetapi mungkin tidak optimal untuk beban yang tidak seimbang.
- IP Hash efektif jika diperlukan konsistensi dalam penentuan server berdasarkan alamat IP.
- Keputusan terbaik tergantung pada karakteristik dan kebutuhan khusus dari aplikasi atau lingkungan sistem yang sedang dihadapi. Jika mungkin, pengujian lebih lanjut dan pemantauan performa di lingkungan produksi dapat memberikan wawasan tambahan.