

Nilai ideal ServerLimit dan MaxClients pada Apache

Written by BiRU

Sunday, 17 January 2016 10:18 -

Pada web server Apache besarnya RAM sangatlah menentukan performa website kita, jadi semakin besar RAM maka website kita jadi bisa semakin banyak menerima pengunjung yang datang secara bersamaan.

Permasalahan yang sering muncul adalah server kekurangan memory yang mengakibatkan Apache harus menggunakan swap memory (disk) yang berakibat loading website kita menjadi sangat lambat.

Misal ada 200 orang yang masuk secara bersamaan padahal daya tampungnya hanya 100 tapi karena kita salah dalam menentukan server limit dan max clients pada httpd.conf maka ke 200 orang tersebut tetap masuk secara bersamaan dan bukannya mengantri per 100 orang.

Secara default Apache mempunyai konfigurasi sbb:

```
StartServers      8 MinSpareServers  5 MaxSpareServers  20 ServerLimit      256
MaxClients        256 MaxRequestsPerChild 4000
```

Yang berarti mendeklarasikan kalau daya tampungnya adalah 256.

Untuk menentukan nilai yang paling pas untuk ServerLimit dan MaxClients adalah sbb:

Total RAM - RAM yang digunakan oleh Linux, MySQL, dll. / rata-rata proses httpd.

Untuk memperoleh jumlah rata-rata RAM yang httpd gunakan bisa mengeksekusi perintah berikut:

```
ps aux | grep 'httpd' | awk '{print $6/1024;}' | awk '{avg += ($1 - avg) / NR;} END {print avg " MB";}'
```

Atau kalau dengan jumlah prosesnya:

```
ps -ylC httpd --sort:rss | awk '{sum+=$8; ++n} END {print "Tot="sum("n");print
```

Nilai ideal ServerLimit dan MaxClients pada Apache

Written by BiRU

Sunday, 17 January 2016 10:18 -

```
"Avg="sum"/"n"="sum/n/1024"MB"}
```

Dan untuk mengetahui jumlah RAM yang MySQL gunakan:

```
ps aux | grep 'mysql' | awk '{print $6/1024 " MB";}'
```

Untuk CMS seperti Drupal, WordPress rata-rata menggunakan 50MB RAM tiap prosesnya, bahkan ada yang lebih kalau terlalu banyak plugin yang digunakan.

Misal kita punya RAM 4GB, tiap proses Apache adalah 50MB, MySQL sejumlah 300MB, dan untuk proses yang lain 200MB. Kita hitung saja:

$4096-500/50=71,92$. Dibulatkan menjadi 70.

Nah kemampuan server kita adalah 70 orang saja, jadi misal ada 100 orang yang masuk secara bersamaan yang 30 orang antri terlebih dahulu.

```
StartServers      8 MinSpareServers  5 MaxSpareServers  20 ServerLimit      70
MaxClients        70 MaxRequestsPerChild 4000
```

Pengaruh loading dengan seperti ini malah lebih terkontrol, misal loading ketika ada 70 orang masuk 1 detik, nah ketika ada 100 orang yang 30 orang menunggu antara 2-3 detik saja.