



## INTISARI

### ***MODIFIKASI RESOURCE MULTICORE SERVER DENGAN KERNEL BYPASS***

Oleh

DIMAS FEBRIYAN PRIAMBODO

15/388467/PPA/04906

Teknologi berkembang sangat cepat ditandai dengan munculnya banyak inovasi baik dari hardware maupun *software*. Perkembangan ini juga berimbas kepada server dengan munculnya *multicore server*. *Multicore server* dengan jumlah *core* yang semakin banyak menuntut *software* yang efisien. *Kernel* dan *Hardware* yang difungsikan menangani berbagai kebutuhan operasional mempunyai beberapa keterbatasan. Keterbatasan ini disebabkan karena tingkat kompleksitas yang tinggi khususnya dalam menangani kebutuhan sebagai peladen seperti *single socket descriptor*, *single IRQ* dan kekurangan dalam *pooling* sehingga memerlukan beberapa modifikasi. Metode *kernel bypass* merupakan salah satu metode untuk mengatasi kekurangan dari *kernel*. Modifikasi pada server ini merupakan gabungan dua sasaran utama yaitu menaikkan *throughput* dan menurunkan *latency server*. Modifikasi yang merupakan gabungan dari modifikasi pada level *driver* dengan *hashing rx signal* dan modifikasi *multiple receiver* dengan *multiple ip receiver*, *multiple thread receiver* dan *multiple port listener* digunakan untuk meningkatkan *throughput*. Modifikasi menggunakan prinsip *pooling* baik pada level *kernel* atau level program digunakan untuk menurunkan *latency*. Gabungan modifikasi ini menjadikan *server* menjadi lebih dapat diandalkan dengan peningkatan *throughput* rata-rata sebesar 250,44 % dan penurunan *latency* 65,83 %. Penerapan di *webserver* memberikan dampak positif dalam meningkatkan HTTP request sebesar 2,99 kali dan penurunan *latency* rata-rata sebesar 20% dengan penurunan *stddev latency* sebesar 88,13% yang menunjukkan *latency* yang stabil.

**Kata Kunci :** *hash rx, multiple ip, multiple port, multiple thread, pooling, bypass kernel*



## ABSTRACT

### ***RESOURCE MODIFICATION ON MULTICORE SERVER WITH KERNEL BYPASS***

By

DIMAS FEBRIYAN PRIAMBODO

15/388467/PPA/04906

Technology is developing very quickly marked by many innovations from both hardware and software. This development also affects the server with multicore server. Multicore server always multiply the number of core, It demanding an efficient software. Kernel and Hardware functional handle various operational needs have some limitations. This limitation is due to the high level of complexity especially in handle as a server such as single socket discriptor, single IRQ and lacking in pooling so it requires some modifications. Bypass kernel is one of the methods to overcome the deficiencies of the kernel. Modifications on this server is a combination of two main goals that increase throughput and lower server latency. Modifications are a combination at the driver level with hashing rx signal and multiple receiver modifications with multiple ip receivers, multiple thread receivers and multiple port listener used to increase throughput. Modifications using pooling principles at either the kernel level or the program level are used to decrease the latency. This combination of modifications makes the server more reliable with an average througput increase of 250.44% and a decrease in latency 65.83%. Implementation on the webserver had a positive impact on increasing HTTP request by 2.99 times and the average latency decrease by 20% with a decrease in stdev latency of 88.13% indicating stable latency

**Keyword :** *hash rx, multiple ip, multiple port, multiple thread, pooling, bypass kernel*