

ABSTRAK

saat ini internet merupakan kebutuhan penting untuk masyarakat luas ataupun perusahaan. Internet sering digunakan untuk sarana berkomunikasi dengan orang lain atau menyimpan data penting di sebuah server melalui website. namun akibat client sebuah website terlalu banyak, server sering mengalami masalah. masalah yang sering dialami server tidak mampu mengirimkan respons yang di request client-nya. Masalah ini terjadi karena setiap server memiliki batas maksimal untuk melayani client.

Sebagai salah satu cara menangani salah satu masalah di atas, penulis memilih untuk melakukan penambahan server dan memasang Load Balancer HAproxy di antara server dan client. Cara kerja pemecahan masalah ini adalah ketika client melakukan request, request akan diterima oleh Load balancer, Loadbalancer akan melakukan request ke server yang aktif sesuai Algoritma yang dipasang pada Load Balancer, kemudian Load Balancer meneruskan respond yang diterima dari server ke client yang mengirimkan request. Pada proyek implementasi ini penulis menggunakan dua algoritma, algoritma round robin dan algoritma source. Hasil dari penerapan algoritma round robin adalah Load Balancer mengirimkan respond ke client mengambil data dari server satu. Jika client melakukan request lagi ke Load Balancer, Load Balancer akan mengambil data ke server dua begitu seterusnya melakukan secara bergantian. Pada algoritma source, Load Balancer akan melakukan request ke salah satu server kemudian data akan dikirim ke client yang melakukan request. Ketika client melakukan request lagi, Loadbalancer akan tetap terhubung ke server yang pertama kali terhubung.

ABSTRACT

now the internet is an important requirement for the wider community or company. The internet is often used for means of communicating with other people or storing important data on a server through a website. but due to the client a website is too much, the server often has problems. the problem that is often experienced by the server is not able to send the response in the client request. This problem occurs because each server has a maximum limit for serving clients.

As one way to handle one of the problems above, author chose to add a server and install the HAProxy Load Balancer between the server and client. The way this problem solving works is when the client makes a request, the request will be received by the Load balancer, Loadbalancer will make a request to the active server according to the Algorithm installed in the Load Balancer, then Load Balancer continues the response received from the server to the client that sends the request. In this implementation project the author uses two algorithms, round robin algorithm and source algorithm. The result of implementing a round robin algorithm is Load Balancer sending a response to the client retrieving data from server one. If the client makes another request to Load Balancer, Load Balancer will take data to server two and so do alternately. In the source algorithm, Load Balancer will make a request to one of the servers then the data will be sent to the client who made the request. When the client requests again, the Loadbalancer will still be connected to the server that was first connected.