

1. Introduction

1.1 Background

In the data center there are multiple servers operated to handle large requests from clients. To manage this request load balancing is needed [1]. Layer-4 load balancer can be used to distribute requests across multiple servers. Layer-4 load balancing makes the request load balancing decision based on information in Layer-4 that is transport layer [2]. Layer-4 in this study refers to the fourth layer of the Open System Interconnection System (OSI). Layer-4 Load Balancer sending a request to corresponding server by using 5-tuple information of such as IP Address Source, IP Address Destination, Port Source, Port Destination, and Transport protocol to determine the server to handle the request coming from client [2].

To build proper Layer-4 Load balancer there are two requirements that need to be met [3] [4]. First is uniformly distributing requests from clients to servers in the data center. Second is ensuring packet forward to the same server from the first packet forwarded to establish a session until the last packet to close a session from client to server or called connection-affinity.

The first requirement that is uniformly distributing the packet is important to split large requests coming from clients and distribute to multiple servers located in a data center to prevent load imbalance. The second requirement is ensuring Connection-affinity [4] [5]. This concept is achieved when the set of packets to establish a session or called flow sent consistently to the same server until the last packet to close the session and not send it to different server. The purpose of ensuring Connection-affinity is to prevent request failure. The IP Hash algorithm is used in load balancing systems such as Hardware, Software, and Software Defined Network. The IP Hash algorithm is used for Layer-4 Load Balancing in data centers because of algorithm capability in distributing request across multiple servers and ensure connection affinity [6] [7] [8].

1.2 Problem statement

However, IP Hash load balancing suffers from load imbalance [3]. IP Hash relies only on hash calculation to determine server index, this can lead to load imbalance.

To mitigate IP Hash Problem WRR algorithm is proposed. The WRR algorithm works by distributing the request in turn for each server but also takes consideration of server capability in handling n number of requests from client [9]. By taking consideration of server capability in handling request, WRR can mitigate load imbalance cause by IP Hash that only depend on hash function to determine server index.

1.3 Purpose

WRR is scheduling algorithm that derives from Round-Robin algorithm, In Round robin the request is forwarded to the server in turn starting from the first server in index to the last server in index. WRR is similar to Round Robin, but the difference in WRR is the weight is assigned to the servers [5]. The Weight is determined by network administrator after observing the capability of servers in handling request, such as link bandwidth from load balancer to the server and/or server specification. To maintain connection affinity in WRR algorithm, Hash function from IP Hash is used and combined with WRR Algorithm.

Implementation of IP Hash and WRR algorithms can be done in hardware, software, or SDN load balancing using F5 hardware, NGINX software, or Openflow Protocol for software-defined networks[6] [8] [10]. But recently there is an emerging paradigm that is Programmable Data Plane. Programmable data plane is a concept that allows programmability of packet processing network directly in data plane [11]. With this concept load balancing can be applied directly in programmable switch without the need for middleware. IP Hash and WRR algorithm can be implemented directly in Programmable switch using P4 (Programming Protocol-independent Packet Processors) [12] programming language.

In this study WRR is proposed to mitigate load imbalance caused by IP Hash algorithm. By using WRR, each server will be assigned weight to prevent load imbalance. Implementation of IP Hash and WRR will be implemented using P4 programming language in BMv2 programmable switch.