

Weekly Report - Week 8

Honam Bang and Jens Emil Gydesen

May 26, 2015

1 Project Summary

We are developing a layer 7 load balancing system, codename “Meercat”, that uses several load balancers to more efficiently balance the traffic amongst servers in a data center. The idea is that, rather than having a single load balancer, we have several load balancers that communicate with each other to load balance all servers. Each load balancer is able to send traffic to any server and we will use statistics to identify the required resources for each request.

We have looked at the Ananta (L4 software load balancer) paper (reference in the Duet paper from class)[1] and also HAProxy[2].

The goal is to, by having multiple load balancers, to make the system very scalable to a high amount of servers as well as eliminating having a single point of failure.

2 Progress

In implementation part, we have implemented a single software load balancer for layer 7 and the traffic test is also done with 'httperf' tool in Linux. By the result, our single load balancer at least can handle amount of http request easily. For the CPU/MEM-bound requests, we also confirmed that our load balancer can pass the test. For now it checks hosts' status by time-based polling in every 5 seconds. So far, we believe that we are ready to go the next step, 'multi level load balancing system' perfectly.

In terms of setting up the test environment, we have by now fully automated Emulab such that we can simply define the number of end-hosts and the number of load balancers and then each load balancer will automatically connect to all of the end-hosts and start balancing load.

3 For Next Week

For next week, we will design a proper topology for multi load balancing system in consideration of our performance measurement experiment. A topology will include 9 10 instances at Emulab cloud. One instance will be used as a traffic generator and 2 3 instances will be used for load balancers, and the remaining instances are for hosts. Also, we will research on our research question: “Why should we have multiple load balancers and why should they communicate with each other?”. For now, we have a plan to focus on the failure-tolerance.

For the implementation part, we start connecting load balancers together. We will design communication protocols and when/how they share some meaningful information and how we will handle a failure at one of the load balancers.

References

- [1] Patel, Parveen, et al. "Ananta: Cloud scale load balancing." ACM SIGCOMM Computer Communication Review. Vol. 43. No. 4. ACM, 2013.
- [2] Tarreau, Willy. "HAProxy-The Reliable, High-Performance TCP/HTTP Load Balancer." 2011-8)[2013-4]. <http://haproxy.lwt.eu> (2012).