Measuring VM’s Performace in AWS, GCP

Network, CPU, Disk I/O measured in stackdriver

Shyam sundar Ramamoorthy – shra3971@colorado.edu

***Abstract*—To compare cpu utilization, network performance between Amazon AWS, Google cloud platform**

I. INTRODUCTION (CLOUD SERVICE PLATFORM)

Amazon Web Serices and Google Compute Engine are both leading cloud service platform. Both offer VM’s in cloud. Network performance is measured using iperf3. Stackdriver offers GUI for viewing network traffic, cpu utilization, Disk IO for a particular interval of time.

II. METRICS FOR MEASUREMENT

A. Zones and Regions of platform

AWS and GCP both offers services in different parts of the world. Different pricing tactics are adopted for hosting VM’s based on memory, number of CPU used in instance. Also price varies for sending traffic between instances across zones and regions. So to avoid high pricing, most of the experiments are carried between instances in same regions.

B. Bandwidth of network

Using iperf as client-server model, traffic is pumped using different parameters like types of traffic (tcp/udp), congestion window size and bandwidth of target instance.

C. Traffic type, Number of Streams, CPU utilization

TCP and UDP are used testing network performance. For tcp traffic, number of parallel streams opened with server is varied. Cpu utilization during all measurements are noted. Most of test are carried out between internal IP address to avoid high cost.

D. Length of Buffer, Congestion Window size

Length of sender buffer is varied and check for packet loss. Also congestion window of network is changed to check for packet loss in the network.

III. PREPARATION

Setup AWS and GCP accounts. Create two instances in same regions. Stackdriver is used to view cpu utilization, number of writes, throughput. All the instances of AWS, GCP are binded with stackdriver. Install iperf3 in all instances created. Write a script for varying bandwidth of network. Python script to pass different arguments and capturing the consolidated output like throughput. Also capture number of packets lost, jitter for udp traffic.

IV. EXPERIMENT

A) Types of VM instances:

AWS has instance types 1) t2.micro- 1(vCPU) 1GB memory, 2) t2.small- 1(vCPU), 2GB memory 3) m4.large- 2(vCPU) 8GB memory. Similar type of instances are chosen in GCP. Fig1 & 2 has bandwidth set for target instance and different amount of data load is sent. Eg: 1M, 1G, 10G data sent via internal IP address for target bandwidth 128K, 128M, 128G respectively and below table is populated.

A. Figures and Tables

Load	**TCP**	**UDP**		
	Bandwidth	***Bandwidth***	***Jitter***	***Packet Loss***
1MB	1.13Mbits/sec	129Kbits/sec	0.102ms	0/127(0%)
1GB	128Mbits/sec	128Mbits/sec	0.080ms	0/131068(0%)
10GB	863Mbits/sec	1.08Gbits/sec	0.094ms	198088/1310631(15%)

Fig1: Amazon AWS t2.micro (Internal IP traffic)

Load	**TCP**	**UDP**		
	Bandwidth	***Bandwidth***	Jitter	***PacketLoss***
1MB	1.13Mbits/sec	129Kbits/sec	0.047ms	0/128(0%)
1GB	128Mbits/sec	128Mbits/sec	0.085ms	1253/131072(0.96%)
10GB	999Mbits/sec	975Mbits/sec	0.171ms	1539/1310720(0.12%)

Fig2: Google Compute Cloud f2.small (Internal IP traffic)

V. OBSERVATION:

Packet loss is seen more in AWS for high bandwidth whereas it is low for GCP. From the below table, it is figured that throughput is consistent in GCP for higher CPU machines. But throughput of instances reduces with higher CPU machines.

Instance Type	**AWS (time interval=10)**		**GCP(time=10)**	
	Data	***Throughput***	***Data***	***Throughput***
t3.micro	1.12 GB	966 Mbits/sec	1.14 GB	980 Mbits/sec
t2.small	1.08 GB	908 Mbits/sec	1.23 GB	1.06 Gbits/sec
m4.large	650 MBytes	545 Mbits/sec	1.65 GB	1.41 Gbits/sec

Fig3: AWS, GCP with external traffic (Traffic send via public IP address)

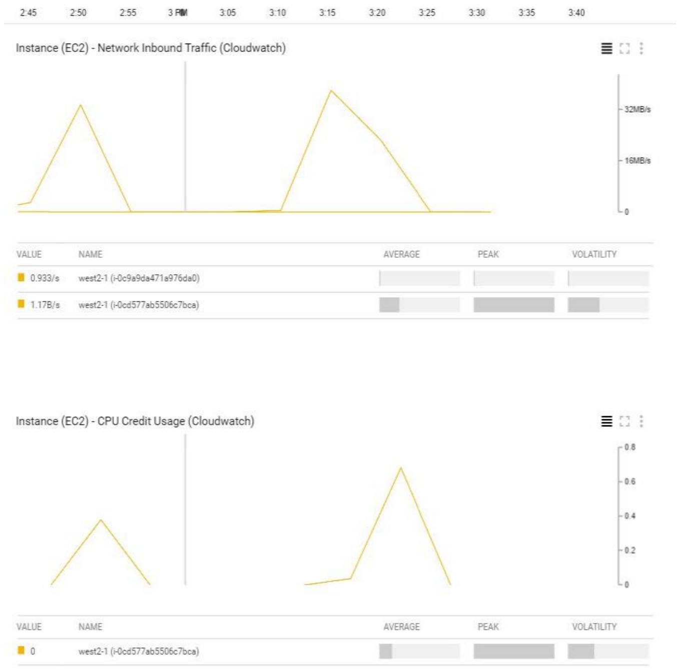

Fig3 : AWS Network & CPU Performance

Fig3 & Fig4 is obtained with stackdriver which is google app. AWS instances are linked with it via IAM role. Both graphs are obtained by sending same amount of data on same type of instances. Here, AWS throughput looks a little less than GCP. But this may occur due to that the AWS instances are connected externally to google app whereas GCP instances connected internally to google app.

Compute engine cpu utilization is high compared to AWS. Also lot of internal traffic goes in GCP instances whereas it is avoided in AWS. Performance of both systems equally performs well in micro instances. Since larger CPU instances are not available for free tier, performance analysis is done less.

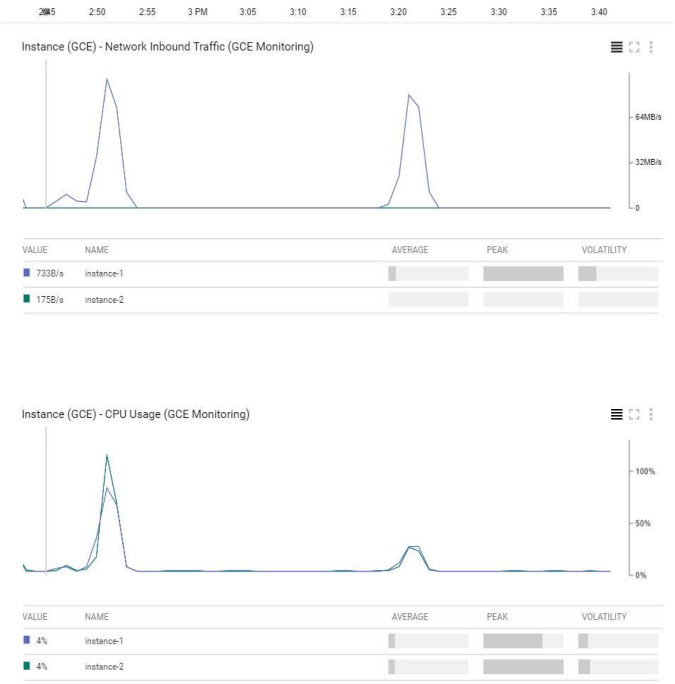

Fig4 : GCP Network & CPU Performance

VI. CONCLUSION:

Even though Google compute engine entered late to market, GCP has more API's compared to AWS. Also, from the observation, GCP wins for higher throughput. For parallel streams (process) AWS wins over GCP. CPU utilization in GCP is going to 65% for single stream where AWS reaches only 60% for two streams.

Overall, GCP wins over AWS in network throughput & usability whereas AWS wins in CPU utilization.