****

**Performance Analysis & Recommendations**

**Provided by**

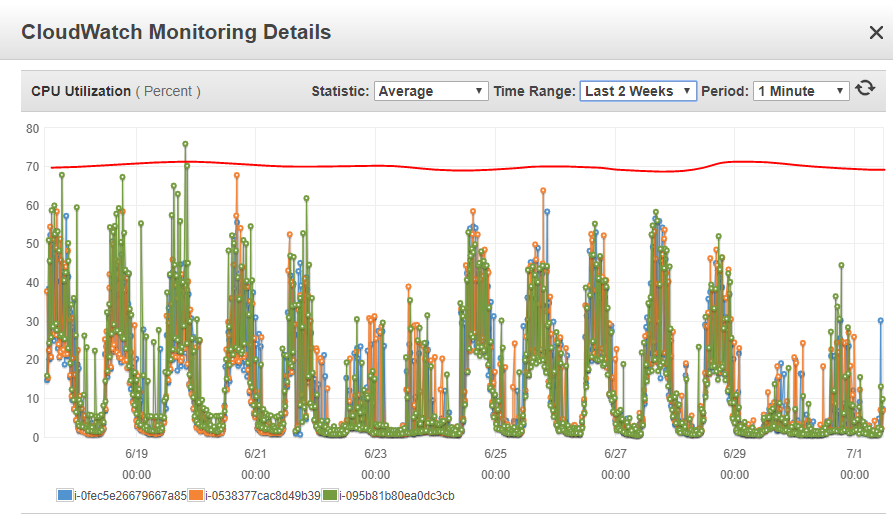
****

**Table of Contents**

1. **Analysis on Portal webservers** 
   1. CPU metric analysis …………………………………… 3
   2. RAM metric analysis …………………………………… 3
   3. Network analysis ……………………………………….. 4
   4. Recommendation ……………………………………... 4
2. **Analysis on WS servers** 
   1. RAM metric analysis …………………………………. 6
   2. CPU metric analysis …………………………………. 6
   3. Network analysis …………………………………….. 6
   4. Recommendation …………………………………….. 6
3. **Analysis on Mobile servers**
   1. CPU metric analysis …………………………………. 7
   2. RAM metric analysis …………………………………. 7
   3. Recommendation …………………………………….. 8
4. **Analysis on External API**
   1. CPU metric analysis …………………………………. 9
   2. RAM metric analysis …………………………………. 9
   3. Recommendation …………………………………….. 9
5. **Analysis on HA-DC1 and HA-DC4**
   1. CPU metric analysis …………………………………. 10
   2. RAM metric analysis …………………………………. 10
   3. Recommendation …………………………………….. 11
6. **Analysis on SQL server**
   1. CPU metric analysis …………………………………. 12
   2. Network analysis …………………………………….. 12
   3. EBS vloumes analysis ……………………………….. 12
   4. Recommendation …………………………………….. 13
7. **Analysis on SQL-PM server**
   1. CPU metric analysis …………………………………. 14
   2. Network analysis …………………………………….. 15
   3. EBS vloumes analysis ………………………………. 15
   4. Recommendation ……………………………………. 15
8. **Conclusion/Overall Recommendation on DB …………… 15-16**

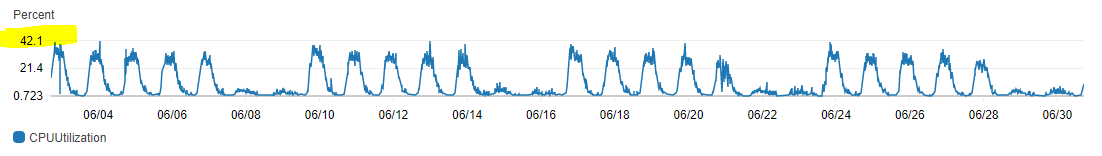
**Analysis on Portal webservers:**

**CPU Metrics for last 2 weeks**



Metrics are indicating CPU load is up and down during the business working hours and they are not crossing 70% in the last 2 weeks.

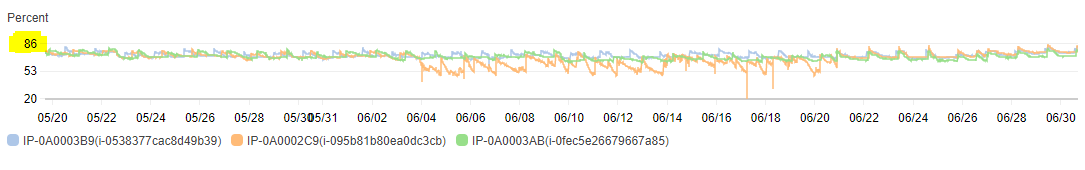
**CPU Metrics by ASG:**

****

**CPU metrics are predictable**

**RAM Metrics for the last 6 weeks**

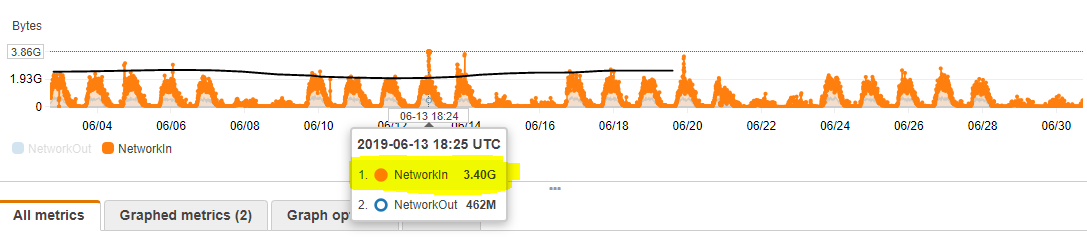
Ram metrics for the last 6 weeks indicates utilization was above 70%



**Network Metrics:**

Portal servers Network metrics are predictable

Metrics indicate **Network out** in control but **Network in** traffic was not in the expected limit.



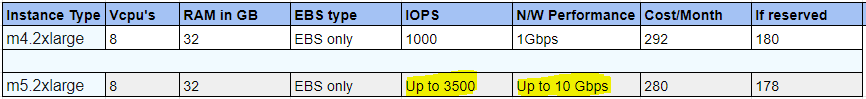
Basic N/W performance for m4 series instances is 1Gb/s.

The above metrics are indicating **Network In** is out of control/boundary. It will affect application performance. During some periods it crossed 3GB/s

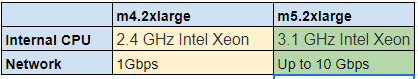
**Recommendation:**

**1.** Portal service related servers are working 90% good but we have an issue with Network. To overcome this we need to update the instance family from **m4** to **m5** series.

**For example**



**Value added:**



**2.** Auto-scaling policies are not configured properly. Min, max, desired capacity are the same.

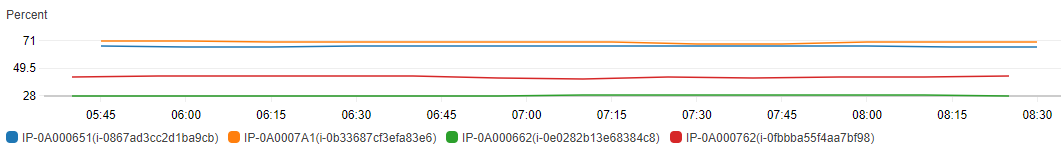
The recommended way is to maintain **Min - 2, Maximum - 4/5, Desired -3**

AutoScaling policies should be adjusted according to the load and traffic. Currently, it has been configured with **scale-out at 17%** and **scale in 11%.**

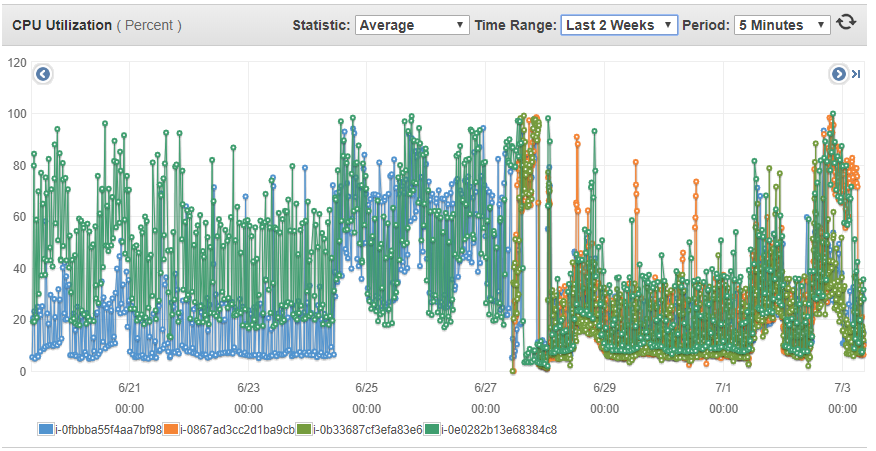
Basically, It should be in between **70%-85%** for **Scale-out** and **10%-20%** for **Scale-in** operations.

**WS servers group analysis:**

**Memory Metrics:**

****

**CPU Metrics**

****

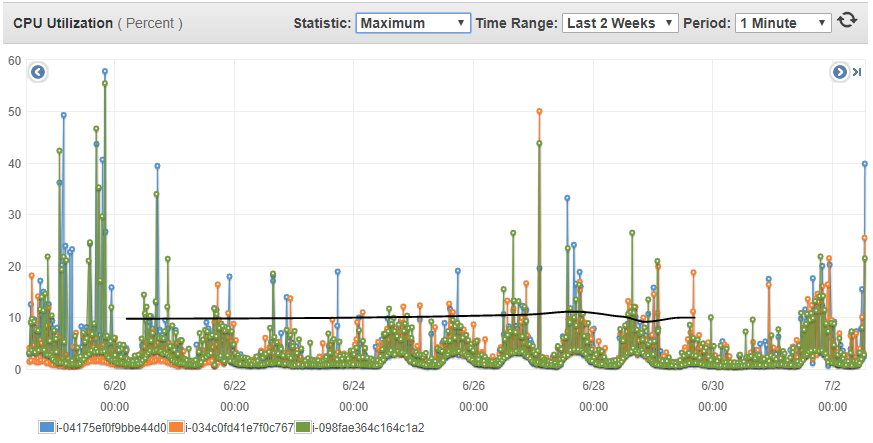
**Recommendation:**

During working hours CPU utilization is increasing but RAM is running constantly. Looks like we need generic capacity like RAM and CPU and Network as these servers are using for batch processing.

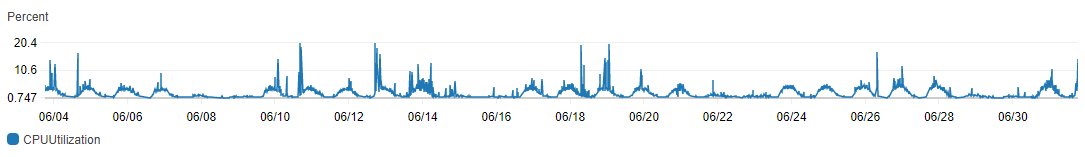
The recommendation will be to choose **m5 series** family. In this servers group, servers are memory optimized and we have less compute capacity and running at max values of boundaries. Better to choose general purppose EC2 family **m5.large** machines to have equal CPU, RAM, N/W and adjust auto scaling policies accordingly.

**Mobile servers Analysis:**

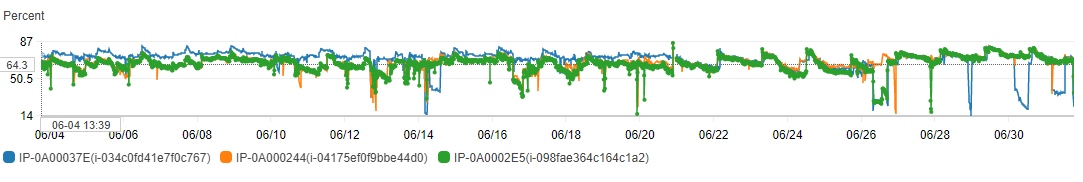
CPU seems fine, running at 10% and reaching peaks only at few instances.



**CPU Metrics by cloudwatch ASG**



Memory Metrics



RAM metrics are not bad, even though they are running in between 65-85%

Network In & Out are also in the expected boundaries.

**Recommendation:**

If we could check more on RAM metrics (used, cached, and free memory), we can decide whether can we remove one server from AG or change the instance model (like m4 family -r4 family)

Better to have ASG policy based on CPU/RAM metric. So that we can adjust the capacity based on the load and reduce the unnecessary spend on the additional instance.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Instance Type** | **VCPU's** | **RAM in GB** | **N/W Performance** | **Cost/Month** | **If reserved** |
| m4.2xlarge | 8 | 32 | 1Gb/s | 292 | 180 |
| **Recommendation** | | | | | |
| r5.xlarge | 4 | 32 | Up to 5Gb/s | 183 | 116 |

**Advantages:**

1. Note: It will reduce CPU capacity. As CPU running at 10% for every time, this instance type can easily handle the load.
2. We will get the same RAM capacity with the advanced memory optimized features
3. Increased N/W performance
4. Reduce $109 on on-demand and $64 if reserved.

**2.** Auto-scaling policies are not configured.

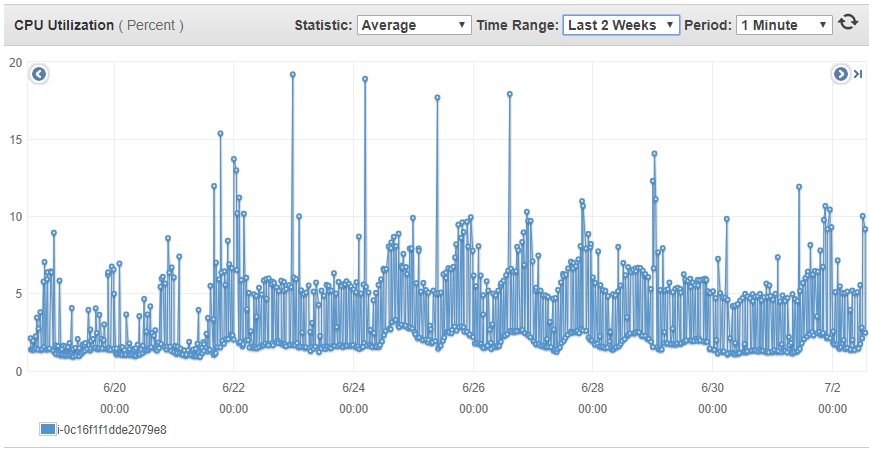
The recommended way is to maintain **Min - 2, Maximum - 4, Desired -3**

AutoScaling policies should be adjusted according to the load and traffic. Currently, there are no auto-scaling policies were configured.

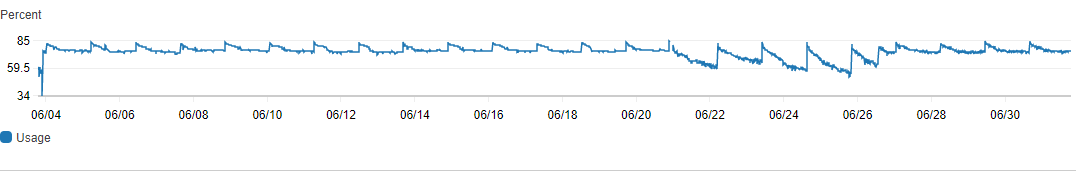
Basically, It should be in between **70%-85% for Scale-out** and **10%-20% for Scale-in** operation. Use RAM metrics for scale-in & out operations instead of CPU metrics.

**Analysis on External API:**

CPU metrics are in expected boundaries



Seems like RAM is utilizing mostly in hard boundaries (Maybe this metric includes cache).



**Recommendation:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Instance Type** | **Vcpu's** | **RAM in GB** | **N/W Performance** | **Cost/Month** | **If reserved** |
| m4.2xlarge | 8 | 32 | 1Gb/s | 292 | 180 |
| **Recommendation** | | | | | |
| r5.xlarge | 4 | 32 | Up to 5Gb/s | 183 | 116 |

**Advantages:**

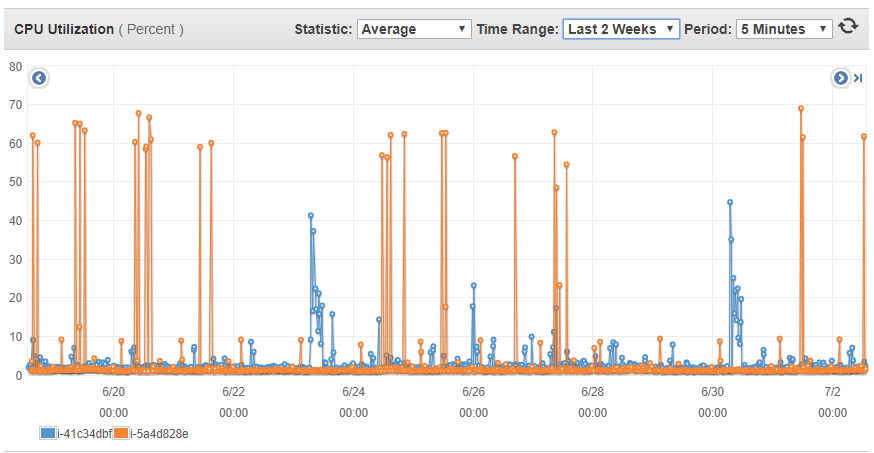
1. Note: It will reduce CPU capacity. As CPU running at 10% for every time, this instance type can easily handle the load.
2. We will get the same RAM capacity with the advanced memory optimized features
3. Increased N/W performance
4. Reduce $109 on on-demand and $64 if reserved.

**Analysis on HA-DC1 and HA-DC4**

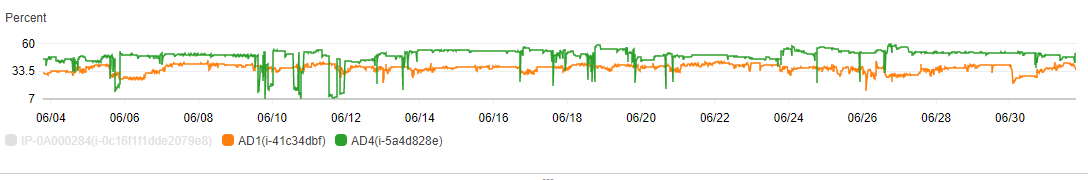
Seems like these two servers utilization was very less and running for internal organization purpose (Might be domain controlers)

**CPU statistics:**

CPU statistics are in the expected boundaries



**RAM Metrics:**

****

**Network Statistics**

Expected boundaries

**Recommendation:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Instance Type** | **Vcpu's** | **RAM in GB** | **N/W Performance** | **Cost/Month** | **If reserved** |
| c3.large | 2 | 3.75 | 500 Mb/s | 76.55 | 53.29 |
| **Recommendation** | | | | | |
| t3.medium | 2 | 4 | Up to 5Gb/s | 30 | 19 |

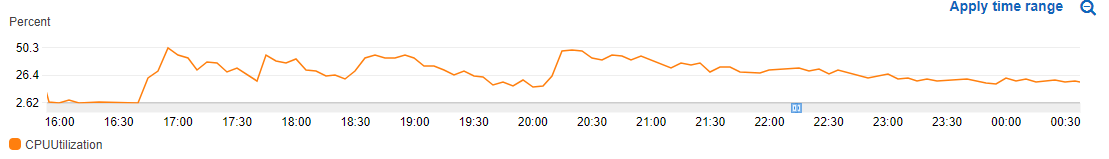
AWS t3 family instances have CPU credits feature. It will automatically burn CPU credit balance whenever there are more requests.

**Advantages:**

1. It will increase 0.25 of RAM with high performance.
2. Increase Network speed.
3. Reduce the cost

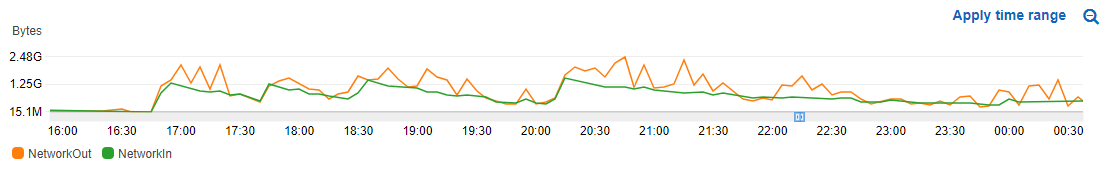
**SQL Server Analysis**

**CPU Utilization:**

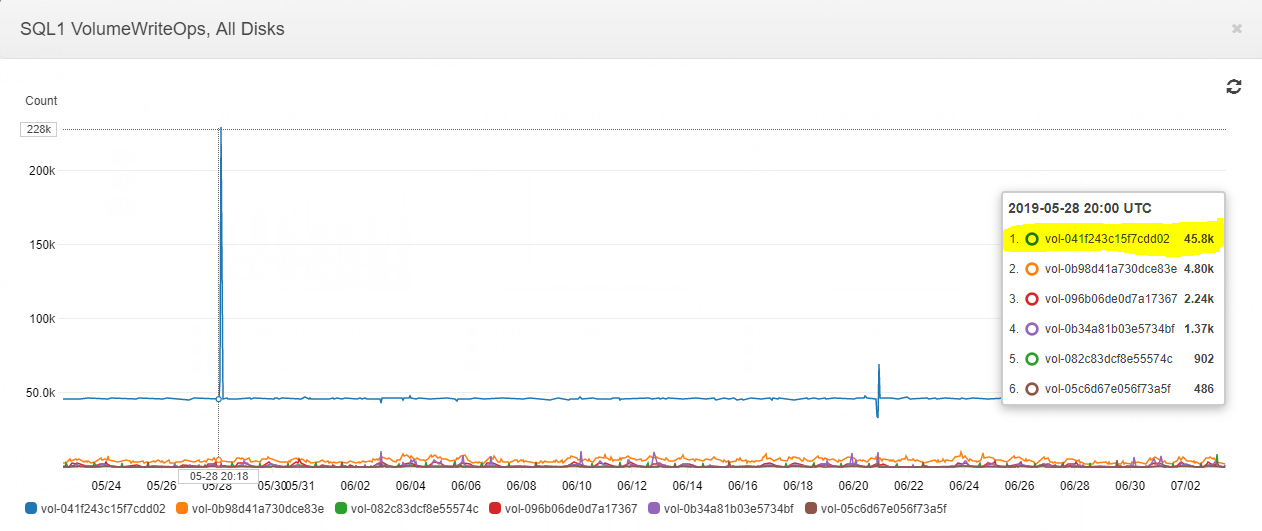
****

CPU utilization is in expected boundaries.

**Network In/Out**

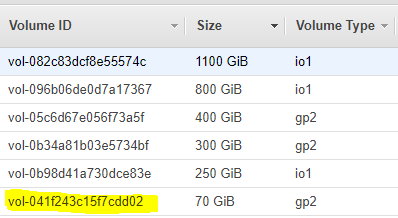
Network statistics are also good

**EBS Volume metrics:**

****

**Recommendation:**

SQL 1 server health is good and metrics are in the expected boundaries but it has an issue with Disk IOPS.

****

The blue bar in the 1st screenshot refers to the above volume. For heavy workloads on databases, we should use the **io1** type of EBS volumes. In cloud watch metrics it was more than the expected boundary (for gp2 max IOPS is 16,000) as it is running highly (above 50,000) it is better to convert it to **io1** and increase IOPS size.

Server health is operating normally and metrics are in the expected boundaries.

Below is the cost calculation for the SQL 1 server

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Details** | **Model** | **Cost** |
| **EC2 Instance** | r4.4xlarge | Memory Optimized | $735 |
| **Volume 1** | vol-082c83dcf8e55574c | io1 | $397 |
| **Volume 2** | vol-096b06de0d7a17367 | io1 | $425 |
| **Volume 3** | vol-05c6d67e056f73a5f | gp2 | $40 |
| **Volume 4** | vol-0b34a81b03e5734bf | gp2 | $30 |
| **Volume 5** | vol-0b98d41a730dce83e | io1 | $128 |
| **Volume 6** | vol-041f243c15f7cdd02 | gp2 | $7 |
|  | | **Total Cost** | **$1,762** |

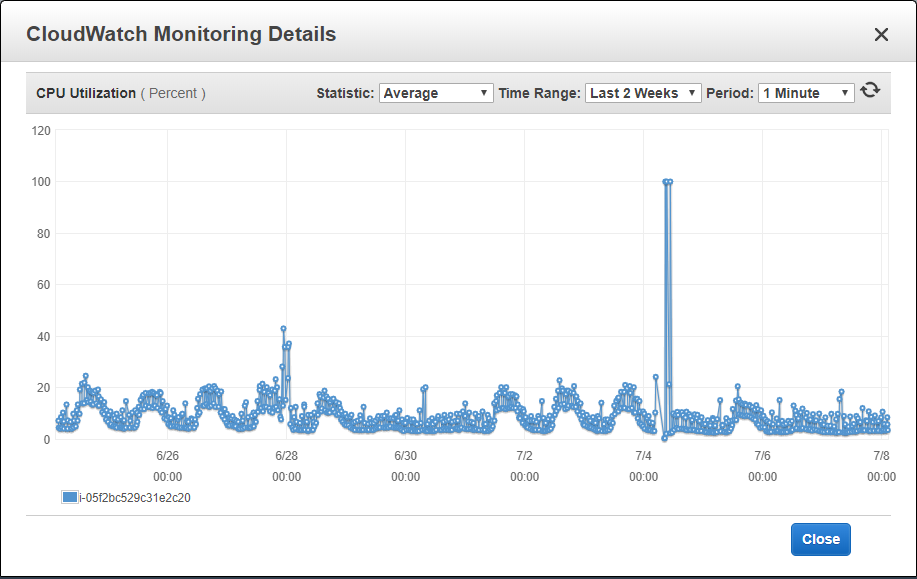
If it is **RDS SQL** with **enterprise edition** we will get **$7400/**month (PAAS model Database)

So the ideal way would be maintaining database in EC2 instance itself. But Volumes which we use, it should be **io1** type with the expected **IOPS** (remind cost.. If we increase more IOPS, we will get additional billing on EBS volumes). With proper IOPS it should run in a great way.

**SQL-PM Analysis:**

**SQL-PM** server metrics also in the expected boundaries.

CPU utilization:

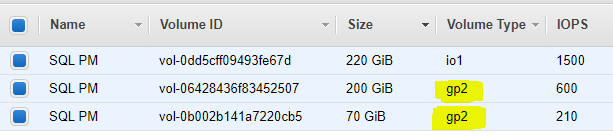


It went to 100% last week during DB servers restart. Usually, it happens when we restart/reboot the instances.

Currently, memory metrics are not available in the cloud watch. Bust as it is memory optimized instance it should be running good as per my expectations.

**EBS Volumes details**

SQL-PM server metrics are in the expected boundaries. But it has complete issue with the EBS volumes only.



**Recommendation:**

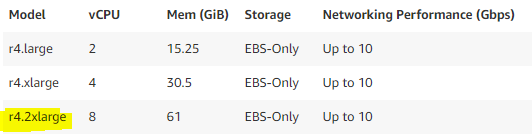
1. According to the EBS volume metrics, there is a Queue length which constantly running
2. It has read and writes latency
3. **Vol-0b002b141a7220cb5** volume has 0 burst balance, it cannot handle the request if some process request data from this EBS volume.

For production environments, we should use io1 EBS volumes at any cost. When we use Database in instances/PAAS model backed EBS volumes should be **io1** with good **IOPS** limits.

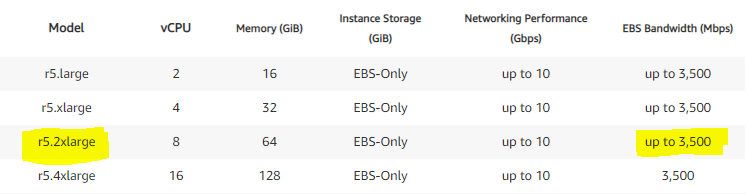
It doesn’t matter how many volumes we use, if an EBS volume is configured for the heavy workload production environments it should be **io1** only.

**Overall recommendation on DB’s:**

**1.** These Database servers are configured with r4 series, we do not have any fixed bandwidth for EBS volumes when we use this series.



If we migrate these two database servers to **r5.2xlarge** we will get following benefits.



We will get EBS Bandwidth up to **3500Mbps.** If we want some fixed bandwidth we can go with **r5a.2xlarge**  also.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Instance Type** | **Vcpu's** | **RAM in GB** | **N/W Performance** | **Cost/Month** | **If reserved** |
| r4.2xlarge | 8 | 61 | Up to 10 Gbps | 388 | 245 |
| **Recommendation** | | | | | |
| r5.2xlarge | 8 | 64 | Up to 10 Gbps | 367 | 232 |

**2.** Update all EBS volume types to **io1** for better performance.

**3.** We need to check what kind of quries we are executing and how frequently we are running them.

**4.** Need to check Any specific querry is taking more time to execute.

**Note: Point - 3&4** can be done only when we have proper DB monitoring.

**5.** If we are running any batch process on production DB, that should be done during non-business working hours.

**6.** If required, need to identify and see whether can we improve querries performance.