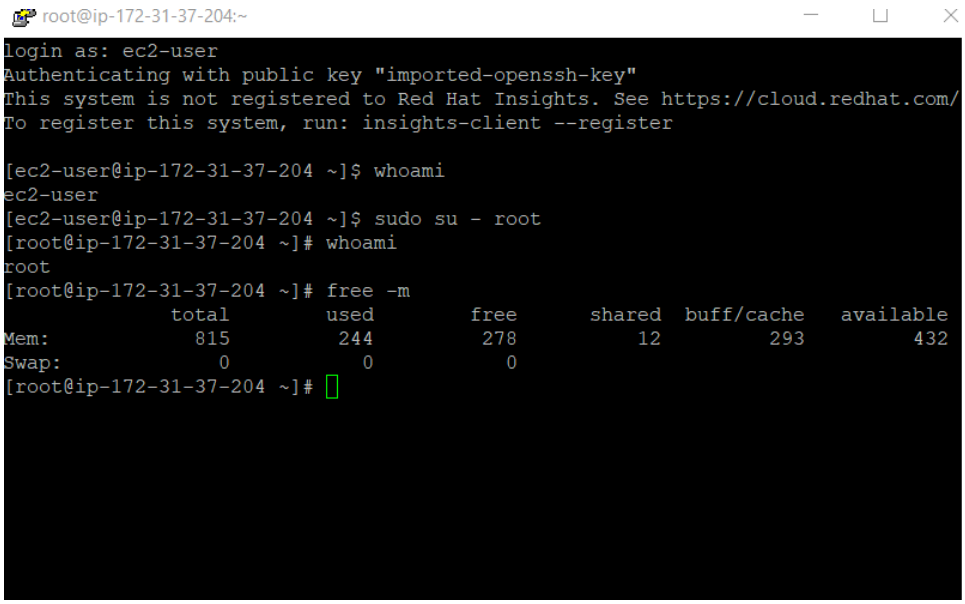
Super Computer with 2+ terabyte storage space:

I want to create a heavy project that includes many fields of computer science, but certainly, my laptop does not have that much storage space. The data that I will need to deploy the project successfully is going to be quite large. Also, my primary goal is that when people check out my project, they don't face any **latency issues**. In previous module, I have created a namenode and datanode using redhat linux operating system. But to increase the efficiency of my system, I need more PCs. So, I decided to launch OS on **AWS cloud**, using the EC2 services. I downloaded multiple instances of redhad linux and made all of them the slave. The storage space at a certain point of time went **beyond 2 terabytes.**

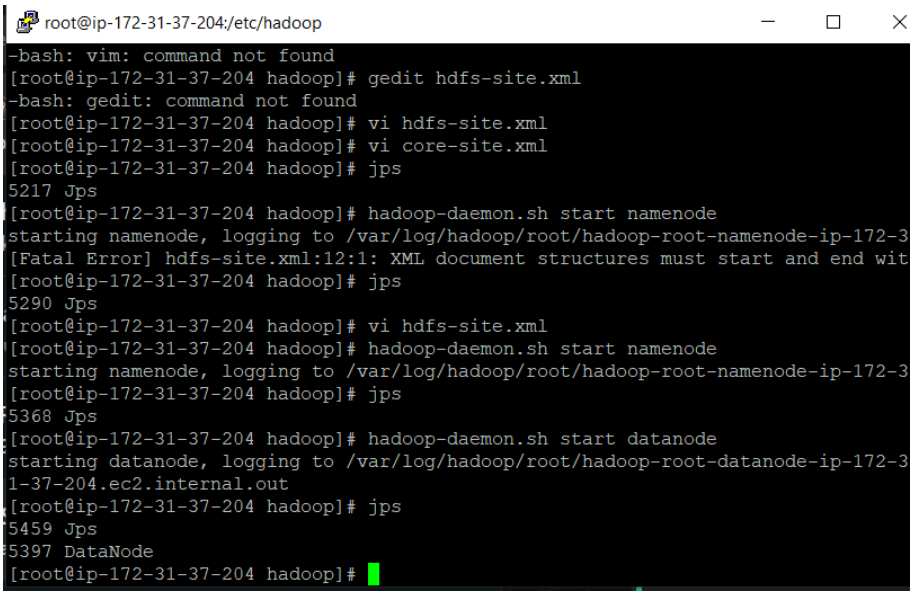
The initial space was as follows:



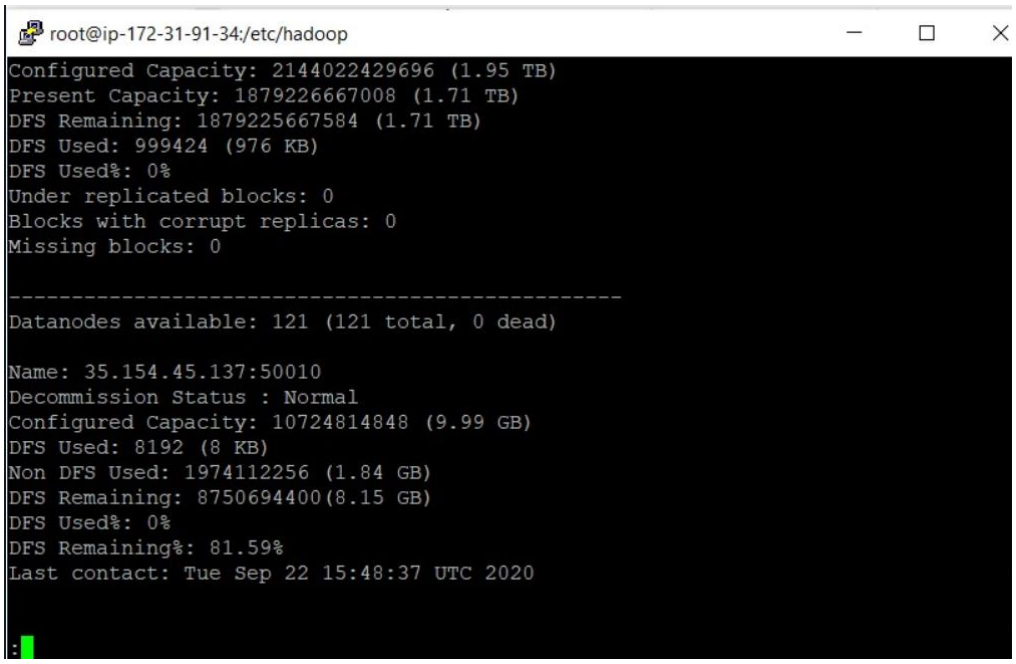
Later I configured the datanode and using the jps command confirmed that it has been started. The steps were the same as shown in the previous module. Same lines were written in the hdfs-site.xml file. The difference was in the core-site.xml file.

The value of the master was:- hdfs://0.0.0.0:9001

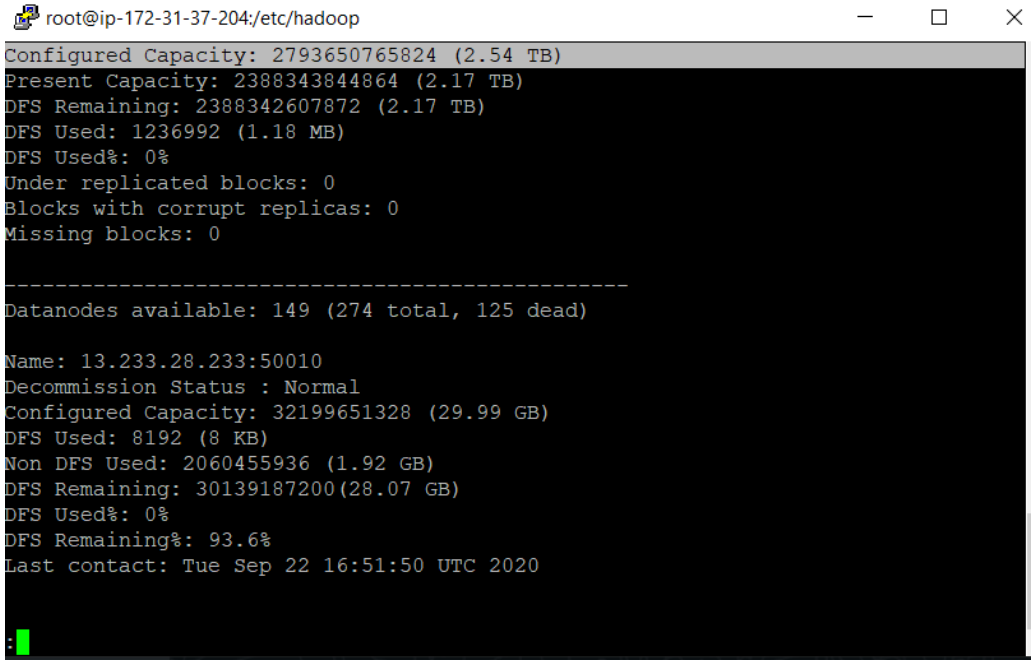
It is because, AWS provides two ip, one public and other private. To enable the datanodes to connect to the master this neutral ip was given because, instance in the cloud only knows the private ip. In the core-site.xml file of the slave, the public ip of the master was given. Rest all steps remains same.



Here we can see that the datanode is connected. Following the similar steps, **100+ datanodes were created.**



Here, 121 datanodes were connected and the memory was around **1.95 TB**. After connecting more slaves, the storage space went up more.



In **total 274 slaves were connected**. The present capacity as seen on the first line of the image is **2.17 TB.**

Now, this supercomputer is ready to deal with **BIGDATA** and solve two prominent problems that is volume and velocity.

Now, the master on receiving the data will strip it into 274 block and send each block to its slave's memory. This will increase the efficiency of the system. The system becomes faster and after exceeding the number of slaves to a number, it is the "super computer".

In the next module, we shall be working on this **"Hadoop cluster"**, to solve several industrial use-cases.