**Abuchi Okeke**

**09/23/2020**

**Big Data Training**

**CLOUD CONCEPT**

Cloud Computing is an IT services provided through cloud.

Type of services

1. IaaS - Infrastructure as a Service
2. PaaS – Platform as a Service
3. SaaS – Software as a Service

Infrastructure as a Service

First stage of infrastructure development. Provides processing environment and can be implemented through different technologies.

* Data
* Application
* Services
* Network
* Database
* Operating System
* Hardware

eg. VM on Google cloud

Platform as a Service

Second stage for testing and running application from user or business side. Provides environment for developing and running applications. Authentication, authorization, session management and metadata are also part of this service.

* Data
* Application

Other components are taken care of by the cloud service provider.

Software as a Service

Most advanced and complex model. Used in running self-contained application. Everything is handled by the service provider. E.g. Google Doc, e-mail, business intelligence, web conference

Major Cloud Providers

1. Amazon Web Services
2. Microsoft Azure
3. Google Cloud
4. Others. IBM, Alibaba etc.

DATA SYSTEM

1. OLTP – Online Transaction Processing
2. OLAP – Online Analytical Processing

**OLTP:** captures and maintains transaction data in a database. Eg RDMS (MySQL, SQL servers), NOSQL (MongoDB, Cassandra). OLTP databases are read, written, and updated frequently

**OLAP:** uses queries for data mining and analysis of large data captured by the OLTP system. Eg. Data warehouse.

Major difference between RDBMS and NOSQL

**RDBMS** – requires schema and it’s not scalable. Relationship exists between the tables

**NOSQL** – is schema less, it’s scalable and there is not relationship between the tables.

OLTP used

Attendance

Storage

ETL used

Data warehouse

OLAP used

Data Analytics

**ETL**: the data from one or more OLTP databases is ingested into OLAP systems through a process called extract, transform, load (ETL). With an ETL tool, users can collect data from several sources and send it to a destination, such as an OLAP data warehouse, where it is queried by analytics and business intelligence tools for insights.

**Scalability**

* Vertical: scale by adding more power(CPU, RAM etc.) to an existing machine. Limited to an extent.
* Horizontal: scale by adding more machine to your pool of resources. Distributed mode (sharding).

Processes

* **Batch Process**: collection of data in batches before processing. The processing happens of blocks of data that have already been stored over a period of time. For example, processing all the transaction that have been performed by a major financial firm in a week. This data contains millions of records for a day that can be stored as a file or record etc.
* **Stream Process**: It is used to query continuous data stream and detect conditions, quickly, within a small time period from the time of receiving the data. The detection time period varies from few milliseconds to minutes.

**HADOOP**

A distributed computing framework (DCF). Stores data, processes, or run applications on a clusters of commodity hardware. Suitable for handling large scale of data (big data). Data and Computation both are distributed among the Nodes (mahines).

Fault Tolerance

Fault tolerance in Hadoop HDFS refers to the working strength of a system in unfavorable conditions and how that system can handle such a situation.

Data divided into Multiple blocks and then have multiple copies on different nodes. So, if any node goes down, data retrieval can be done through other nodes. In this way fault tolerance is achieved in Hadoop

Using Replication Factor, we can achieve to make multiple block into data node. By default, the replication factor is 3 in HDFS. But you can increase the replication as per your requirement. The default is 3 because of its low probability at 0.45

Hadoop System

* **Processing**: Uses map reduce and YARN (Yet Another Resource Negotiator)
* **Storage:** HDFS (Hadoop Distributed File System)

System/File Storage Format

* Windows: NTFS, FAT32 (outdated)
* UNIX: Ext2, Ext3, Ext4 (Ext2 stands for second extended file system. Ext3 stands for third extended file system. Ext4 stands for fourth extended file system). Stores in blocks

HDFS

For Hadoop 2.x>, default block is 128MB whereas for Hadoop 1.x is 64MB

HADOOP FILE FORMAT

1. Name node (NN): stores meta data
2. Data node (DN): stores physical data
3. Secondary NN (SNN): for check pointing
4. Resources manager (RM): for resource allocation
5. Node manager (NM): for task execution

HADOOP ARCHITECTURE

Hadoop HDFS follows the master-slave architecture where the NameNode is the master node and maintains the filesystem tree.

N.N

S.N.N

R.M

D.N, NM

D.N, NM

D.N, NM

* SNN sends updated meta data to NN
* SNN performs checkpointing process.
* Checkpointing process merges edit log file (EDF, data stored in the disk) with FS image file (data stored in the memory)
* Node Manager is responsible for executing the tasks on the shared machine.
* R.M (job tracker) and N.M (task tracker) are components of YARN – Hadoop 1.x.
* N.N, D.N, S.N are components of Hadoop.

High Availability

High Availability was a new feature added to Hadoop 2.x to solve the Single point of failure problem in the older versions of Hadoop. The high availability feature in Hadoop ensures the availability of the Hadoop cluster without any downtime, even in unfavorable conditions like NameNode failure, DataNode failure, machine crash, etc.

Implementation of NameNode High Availability architecture Using **Quorum Journal Nodes**

Lock

DN1

DN3

DN2

ZooKeeper

NameNode1

(Active)

**Quorum Journal Nodes**

**N1, N2, N3**

NameNode2

(Passive/Standby)

* Zookeeper gives the active N.N1 a lock called active fail over controller
* The active node and the passive nodes communicate with a group of separate daemons called “JournalNodes,” which is the lightweight process to get sync with each other.
* The active node writes the edit log modification to the majority of JournalNodes.
* There are generally three JournalNode daemons that allow the system to tolerate the failure of a single machine.
* The system can tolerate at most (N-1) / 2 failures when running with N JournalNodes.
* One should run an odd number of JNs, to increase the number of failures the system tolerates.
* The active NameNode updates the edit log in the JNs.
* The standby nodes continuously watch the JNs for edit log change. Standby nodes read the changes in edit logs and apply them to their namespace.

SECURITY

**KERBEROS** (means a dog with three head in Greek).

Hadoop uses Kerberos as the basis for strong authentication and identity propagation for both user and services. Kerberos is a third-party authentication mechanism, in which users and services rely on a third party - the Kerberos server - to authenticate each to the other. The Kerberos server itself is known as the Key Distribution Center, or KDC. At a high level, it has three parts:

1. A database of the users and services (known as principals) that it knows about and their respective Kerberos passwords
2. An Authentication Server (AS) which performs the initial authentication and issues a Ticket Granting Ticket (TGT)
3. A Ticket Granting Server (TGS) that issues subsequent service tickets based on the initial TGT

A user principal requests authentication from the AS. The AS returns a TGT that is encrypted using the user principal's Kerberos password, which is known only to the user principal and the AS. The user principal decrypts the TGT locally using its Kerberos password, and from that point forward, until the ticket expires, the user principal can use the TGT to get service tickets from the TGS. Service tickets are what allow a principal to access various services.

Because cluster resources (hosts or services) cannot provide a password each time to decrypt the TGT, they use a special file, called a keytab, which contains the resource principal's authentication credentials. The set of hosts, users, and services over which the Kerberos server has control is called a realm.

**Client**

**(Username)**

Ticket

**Service**

**Ticket Generating**

**Server**

**Authenticating server**

Tools Available in Hadoop

* SQOOP: for ingesting data (structured data) from RDMS to HDFS
* HIVE: data warehouse system in Hadoop for online analytical processing (OLAP)
* FLUME: an ingestion tool in near real time (streaming data)
* PIG (PIG Latin): for data transformation and analytics.
* HBASE: used for online transaction processing. It’s non-relational database system. Distributed column-oriented key-value system. Leverages fault tolerance (ACID-atomicity, consistency, isolation, durability Transaction).

Type of Data in HADOOP

* Structured Data: has some kind of schema e.g. RDBMS data
* Semi-Structured Data e.g. CSV, JSON, XML
* Un-structured Data e.g. Audio, video, txt file etc.

All type of data can be stored in HDFS