

## Looking for Real Exam Questions for IT Certification Exams!

**We** guarantee you can pass any IT certification exam at your first attempt with just 10-12 hours study of our guides.

**O**ur study guides contain actual exam questions; accurate answers with detailed explanation verified by experts and all graphics and drag-n-drop exhibits shown just as on the real test.

**T**o test the quality of our guides, you can download the one-fourth portion of any guide from <http://www.certificationking.com> absolutely free. You can also download the guides for retired exams that you might have taken in the past.

**F**or pricing and placing order, please visit <http://certificationking.com/order.html>

We accept all major credit cards through [www.paypal.com](http://www.paypal.com)

**F**or other payment options and any further query, feel free to mail us at [info@certificationking.com](mailto:info@certificationking.com)

**Question No : 1**

Which two updates occur when a client application opens a stream to begin a file write on a cluster running MapReduce v1 (MRv1)?

- A. Once the write stream closes on the DataNode, the DataNode immediately initiates a block report to the NameNode.
- B. The change is written to the NameNode disk.
- C. The metadata in the RAM on the NameNode is flushed to disk.
- D. The metadata in RAM on the NameNode is flushed disk.
- E. The metadata in RAM on the NameNode is updated.
- F. The change is written to the edits file.

**Answer: E,F**

**Question No : 2**

For a MapReduce job, on a cluster running MapReduce v1 (MRv1), what's the relationship between tasks and task templates?

- A. There are always at least as many task attempts as there are tasks.
- B. There are always at most as many tasks attempts as there are tasks.
- C. There are always exactly as many task attempts as there are tasks.
- D. The developer sets the number of task attempts on job submission.

**Answer: A**

**Question No : 3**

What action occurs automatically on a cluster when a DataNode is marked as dead?

- A. The NameNode forces re-replication of all the blocks which were stored on the dead DataNode.
- B. The next time a client submits job that requires blocks from the dead DataNode, the JobTracker receives no heart beats from the DataNode. The JobTracker tells the NameNode that the DataNode is dead, which triggers block re-replication on the cluster.
- C. The replication factor of the files which had blocks stored on the dead DataNode is temporarily reduced, until the dead DataNode is recovered and returned to the cluster.
- D. The NameNode informs the client which write the blocks that are no longer available;

the client then re-writes the blocks to a different DataNode.

**Answer: A**

**Explanation:** How NameNode Handles data node failures?

NameNode periodically receives a Heartbeat and a Blockreport from each of the DataNodes in the cluster. Receipt of a Heartbeat implies that the DataNode is functioning properly. A Blockreport contains a list of all blocks on a DataNode. When NameNode notices that it has not received a heartbeat message from a data node after a certain amount of time, the data node is marked as dead. Since blocks will be under replicated the system begins replicating the blocks that were stored on the dead datanode. The NameNode Orchestrates the replication of data blocks from one datanode to another. The replication data transfer happens directly between datanodes and the data never passes through the namenode.

Note: If the Name Node stops receiving heartbeats from a Data Node it presumes it to be dead and any data it had to be gone as well. Based on the block reports it had been receiving from the dead node, the Name Node knows which copies of blocks died along with the node and can make the decision to re-replicate those blocks to other Data Nodes. It will also consult the Rack Awareness data in order to maintain the two copies in one rack, one copy in another rack replica rule when deciding which Data Node should receive a new copy of the blocks.

Reference: 24 Interview Questions & Answers for Hadoop MapReduce developers, How NameNode Handles data node failures'

#### **Question No : 4**

How does the NameNode know DataNodes are available on a cluster running MapReduce v1 (MRv1)

- A.** DataNodes listed in the dfs.hosts file. The NameNode uses as the definitive list of available DataNodes.
- B.** DataNodes heartbeat in the master on a regular basis.
- C.** The NameNode broadcasts a heartbeat on the network on a regular basis, and DataNodes respond.
- D.** The NameNode send a broadcast across the network when it first starts, and DataNodes respond.

**Answer: B**

**Explanation:** How NameNode Handles data node failures?

NameNode periodically receives a Heartbeat and a Blockreport from each of the DataNodes in the cluster. Receipt of a Heartbeat implies that the DataNode is functioning properly. A Blockreport contains a list of all blocks on a DataNode. When NameNode notices that it has not received a heartbeat message from a data node after a certain amount of time, the data node is marked as dead. Since blocks will be under replicated the system begins replicating the blocks that were stored on the dead datanode. The NameNode Orchestrates the replication of data blocks from one datanode to another. The replication data transfer happens directly between datanodes and the data never passes through the namenode.

Reference: 24 Interview Questions & Answers for Hadoop MapReduce developers, How NameNode Handles data node failures?

#### Question No : 5

Which three distcp features can you utilize on a Hadoop cluster?

- A.** Use distcp to copy files only between two clusters or more. You cannot use distcp to copy data between directories inside the same cluster.
- B.** Use distcp to copy HBase table files.
- C.** Use distcp to copy physical blocks from the source to the target destination in your cluster.
- D.** Use distcp to copy data between directories inside the same cluster.
- E.** Use distcp to run an internal MapReduce job to copy files.

**Answer: B,D,E**

**Explanation:**

DistCp (distributed copy) is a tool used for large inter/intra-cluster copying. It uses Map/Reduce to effect its distribution, error handling and recovery, and reporting. It expands a list of files and directories into input to map tasks, each of which will copy a partition of the files specified in the source list. Its Map/Reduce pedigree has endowed it with some quirks in both its semantics and execution.

Reference: Hadoop DistCp Guide

**Question No : 6**

How does HDFS Federation help HDFS Scale horizontally?

- A.** HDFS Federation improves the resiliency of HDFS in the face of network issues by removing the NameNode as a single-point-of-failure.
- B.** HDFS Federation allows the Standby NameNode to automatically resume the services of an active NameNode.
- C.** HDFS Federation provides cross-data center (non-local) support for HDFS, allowing a cluster administrator to split the Block Storage outside the local cluster.
- D.** HDFS Federation reduces the load on any single NameNode by using the multiple, independent NameNode to manage individual parts of the filesystem namespace.

**Answer: D**

**Explanation:** HDFS FederationIn order to scale the name service horizontally, federation uses multiple independent Namenodes/Namespace. The Namenodes are federated, that is, the Namenodes are independent and don't require coordination with each other. The datanodes are used as common storage for blocks by all the Namenodes. Each datanode registers with all the Namenodes in the cluster. Datanodes send periodic heartbeats and block reports and handles commands from the Namenodes.

Reference: Apache Hadoop 2.0.2-alpha

<http://hadoop.apache.org/docs/current/>

**Question No : 7**

Choose which best describe a Hadoop cluster's block size storage parameters once you set the HDFS default block size to 64MB?

- A.** The block size of files in the cluster can be determined as the block is written.
- B.** The block size of files in the Cluster will all be multiples of 64MB.
- C.** The block size of files in the duster will all at least be 64MB.
- D.** The block size of files in the cluster will all be the exactly 64MB.

**Answer: D**

**Explanation:**

Note: What is HDFS Block size? How is it different from traditional file system block size?

In HDFS data is split into blocks and distributed across multiple nodes in the cluster. Each block is typically 64Mb or 128Mb in size. Each block is replicated multiple times. Default is to replicate each block three times. Replicas are stored on different nodes. HDFS utilizes the local file system to store each HDFS block as a separate file. HDFS Block size can not be compared with the traditional file system block size.

#### Question No : 8

Which MapReduce daemon instantiates user code, and executes map and reduce tasks on a cluster running MapReduce v1 (MRv1)?

- A. NameNode
- B. DataNode
- C. JobTracker
- D. TaskTracker
- E. ResourceManager
- F. ApplicationMaster
- G. NodeManager

**Answer: D**

**Explanation:** A TaskTracker is a slave node daemon in the cluster that accepts tasks (Map, Reduce and Shuffle operations) from a JobTracker. There is only One Task Tracker process run on any hadoop slave node. Task Tracker runs on its own JVM process. Every TaskTracker is configured with a set of slots, these indicate the number of tasks that it can accept. The TaskTracker starts a separate JVM processes to do the actual work (called as Task Instance) this is to ensure that process failure does not take down the task tracker. The TaskTracker monitors these task instances, capturing the output and exit codes. When the Task instances finish, successfully or not, the task tracker notifies the JobTracker. The TaskTrackers also send out heartbeat messages to the JobTracker, usually every few minutes, to reassure the JobTracker that it is still alive. These message also inform the JobTracker of the number of available slots, so the JobTracker can stay up to date with where in the cluster work can be delegated.

Note: How many Daemon processes run on a Hadoop system?

Hadoop is comprised of five separate daemons. Each of these daemon run in its own JVM.

Following 3 Daemons run on Master

nodes NameNode - This daemon stores and maintains the metadata for HDFS.

Secondary NameNode - Performs housekeeping functions for the NameNode.

JobTracker - Manages MapReduce jobs, distributes individual tasks to machines running the Task Tracker.

Following 2 Daemons run on each Slave nodes

DataNode – Stores actual HDFS data blocks.

TaskTracker - Responsible for instantiating and monitoring individual Map and Reduce tasks.

Reference: 24 Interview Questions & Answers for Hadoop MapReduce developers, What is a Task Tracker in Hadoop? How many instances of TaskTracker run on a Hadoop Cluster

#### Question No : 9

What two processes must you do if you are running a Hadoop cluster with a single NameNode and six DataNodes, and you want to change a configuration parameter so that it affects all six DataNodes.

- A. You must restart the NameNode daemon to apply the changes to the cluster
- B. You must restart all six DataNode daemons to apply the changes to the cluster.
- C. You don't need to restart any daemon, as they will pick up changes automatically.
- D. You must modify the configuration files on each of the six DataNode machines.
- E. You must modify the configuration files on only one of the DataNode machine
- F. You must modify the configuration files on the NameNode only. DataNodes read their configuration from the master nodes.

**Answer: B,D**

#### Question No : 10

Identify the function performed by the Secondary NameNode daemon on a cluster

configured to run with a single NameNode.

- A.** In this configuration, the Secondary NameNode performs a checkpoint operation on the files by the NameNode.
- B.** In this configuration, the Secondary NameNode is standby NameNode, ready to failover and provide high availability.
- C.** In this configuration, the Secondary NameNode performs deal-time backups of the NameNode.
- D.** In this configuration, the Secondary NameNode servers as alternate data channel for clients to reach HDFS, should the NameNode become too busy.

**Answer: A**

**Explanation:** The term "secondary name-node" is somewhat misleading. It is not a name-node in the sense that data-nodes cannot connect to the secondary name-node, and in no event it can replace the primary name-node in case of its failure.

The only purpose of the secondary name-node is to perform periodic checkpoints. The secondary name-node periodically downloads current name-node image and edits log files, joins them into new image and uploads the new image back to the (primary and the only) name-node.

So if the name-node fails and you can restart it on the same physical node then there is no need to shutdown data-nodes, just the name-node need to be restarted. If you cannot use the old node anymore you will need to copy the latest image somewhere else. The latest image can be found either on the node that used to be the primary before failure if available; or on the secondary name-node. The latter will be the latest checkpoint without subsequent edits logs, that is the most recent name space modifications may be missing there. You will also need to restart the whole cluster in this case.

Reference: Hadoop Wiki, What is the purpose of the secondary name-node?

#### Question No : 11

You install Cloudera Manager on a cluster where each host has 1 GB of RAM. All of the services show their status as concerning. However, all jobs submitted complete without an error.

Why is Cloudera Manager showing the concerning status KM the services?



- A. A slave node's disk ran out of space
- B. The slave nodes, haven't sent a heartbeat in 60 minutes
- C. The slave nodes are swapping.
- D. DataNode service instance has crashed.

**Answer: B**

**Explanation:** Concerning: There is an irregularity in the status of a service instance or role instance, but Cloudera Manager calculates that the instance might recover. For example, if the number of missed heartbeats exceeds a configurable threshold, the health status becomes Concerning. Or, if an instance is running on a host and the host is rebooted, the instance will be reported as In Progress for some period of time while it is restarting. Because the instance is expected to be Started, its health will be reported as Concerning until it transitions to started.

Note:

Bad: The service instance or role instance is not performing or did not finish performing the last command as expected, and Cloudera Manager calculates that the instance will not recover. For example, if the number of missed heartbeats exceeds a second (higher) configurable threshold, the health status becomes Bad. Another example of bad health is if a role you have stopped is actually still running, or a started role has stopped unexpectedly.

Good: The service instance or role instance is performing or has finished performing the last command as expected. This does not necessarily mean the service is running, it means it is behaving as expected. For example, if you clicked Stop to stop a role instance and it stopped successfully, then that role instance has a Good health status, even though it is not running.

Reference: About Service, Role, and Host Health

#### **Question No : 12**

What is the recommended disk configuration for slave nodes in your Hadoop cluster with 6 x 2 TB hard drives?

- A. RAID 10
- B. JBOD
- C. RAID 5

**D. RAID 1+0****Answer: B****Explanation:**

Note: Let me be clear here...there are absolutely times when using a Enterprise-class storage device makes perfect sense. But for Hadoop it is very much unnecessary, and it is these three areas that I am going to hit as well as some others that I hope will demonstrate that Hadoop works best with inexpensive, internal storage in JBOD mode. Some of you might say "if you lose a disk in a JBOD configuration, you're toast...you lose everything". While this might be true, with Hadoop, it isn't. Not only do you have the benefit that JBOD gives you in speed, you have the benefit that Hadoop Distributed File System (HDFS) negates this risk. HDFS basically creates three copies of the data. This is a very robust way to guard against data loss due to a disk failure or node outage, so you can eliminate the need for performance-reducing RAID.

Reference: Hadoop and Storage Area Networks

**Question No : 13**

You configure you cluster with HDFS High Availability (HA) using Quorum-Based storage. You do not implement HDFS Federation.

What is the maximum number of NameNodes daemon you should run on you cluster in order to avoid a "split-brain" scenario with your NameNodes?

- A.** Unlimited. HDFS High Availability (HA) is designed to overcome limitations on the number of NameNodes you can deploy.
- B.** Two active NameNodes and one Standby NameNode
- C.** One active NameNode and one Standby NameNode
- D.** Two active NameNodes and two Standby NameNodes

**Answer: C**

**Explanation:** In a typical HA cluster, two separate machines are configured as NameNodes. At any point in time, one of the NameNodes is in an Active state, and the other is in a Standby state. The Active NameNode is responsible for all client operations in the cluster, while the Standby is simply acting as a slave, maintaining enough state to provide a fast failover if necessary.

Note: It is vital for the correct operation of an HA cluster that only one of the NameNodes be active at a time. Otherwise, the namespace state would quickly diverge between the two, risking data loss or other incorrect results. In order to ensure this property and prevent the so-called "split-brain scenario," the JournalNodes will only ever allow a single NameNode to be a writer at a time. During a failover, the NameNode which is to become active will simply take over the role of writing to the JournalNodes, which will effectively prevent the other NameNode from continuing in the Active state, allowing the new Active NameNode to safely proceed with failover.

Reference: Cloudera CDH4 High Availability Guide, Quorum-based Storage

**Question No : 14**

You configure Hadoop cluster with both MapReduce frameworks, MapReduce v1 (MRv1) and MapReduce v2 (MRv2/YARN). Which two MapReduce (computational) daemons do you need to configure to run on your master nodes?

- A. JobTracker
- B. ResourceManager
- C. ApplicationMaster
- D. JournalNode
- E. NodeManager

**Answer: A,B**

**Explanation:**

<http://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html>

As you can see, ApplicationMaster is in slave nodes instead of master nodes. So it won't be the answer.

Only JobTracker and ResourceManger are MapReduce daemons running in master nodes so the answers

**Question No : 15**

You observe that the number of spilled records from map tasks for exceeds the number of map output records. Your child heap size is 1 GB and your `io.sort.mb` value is set to 100MB. How would you tune your `io.sort.mb` value to achieve maximum memory to disk I/O ratio?

- A. Tune `io.sort.mb` value until you observe that the number of spilled records equals (or is as close to equals) the number of map output records.
- B. Decrease the `io.sort.mb` value below 100MB.
- C. Increase the `io.sort.mb` as high you can, as close to 1GB as possible.
- D. For 1GB child heap size an `io.sort.mb` of 128MB will always maximum memory to disk I/O.

**Answer: A**

**Explanation:** here are a few tradeoffs to consider.

1. the number of seeks being done when merging files. If you increase the merge factor too high, then the seek cost on disk will exceed the savings from doing a parallel merge (note that OS cache might mitigate this somewhat).

2. Increasing the sort factor decreases the amount of data in each partition. I believe the number is `io.sort.mb / io.sort.factor` for each partition of sorted data. I believe the general rule of thumb is to have `io.sort.mb = 10 * io.sort.factor` (this is based on the seek latency of the disk on the transfer speed, I believe. I'm sure this could be tuned better if it was your bottleneck. If you keep these in line with each other, then the seek overhead from merging should be minimized

3. you increase `io.sort.mb`, then you increase memory pressure on the cluster, leaving less memory available for job tasks. Memory usage for sorting is `mapper tasks * io.sort.mb` -- so you could find yourself causing extra GCs if this is too high

Essentially,

If you find yourself swapping heavily, then there's a good chance you have set the sort factor too high.

If the ratio between `io.sort.mb` and `io.sort.factor` isn't correct, then you may need to change `io.sort.mb` (if you have the memory) or lower the sort factor.

If you find that you are spending more time in your mappers than in your reducers, then you may want to increase the number of map tasks and decrease the sort factor (assuming

there is memory pressure).

Reference: How could I tell if my hadoop config parameter io.sort.factor is too small or too big?

<http://stackoverflow.com/questions/8642566/how-could-i-tell-if-my-hadoop-config-parameter-io-sort-factor-is-too-small-or-to>

#### Question No : 16

Your Hadoop cluster has 25 nodes with a total of 100 TB (4 TB per node) of raw disk space allocated HDFS storage. Assuming Hadoop's default configuration, how much data will you be able to store?

- A. Approximately 100TB
- B. Approximately 25TB
- C. Approximately 10TB
- D. Approximately 33 TB

**Answer: D**

**Explanation:** In default configuration there are total 3 copies of a datablock on HDFS, 2 copies are stored on datanodes on same rack and 3rd copy on a different rack.

Reference: 24 Interview Questions & Answers for Hadoop MapReduce developers, How the HDFS Blocks are replicated?

#### Question No : 17

You set up the Hadoop cluster using NameNode Federation. One NameNode manages the/users namespace and one NameNode manages the/data namespace. What happens when client tries to write a file to /reports/myreport.txt?

- A. The file successfully writes to /users/reports/myreports/myreport.txt.
- B. The client throws an exception.
- C. The file successfully writes to /report/myreport.txt. The metadata for the file is managed by the first NameNode to which the client connects.
- D. The file writes fails silently; no file is written, no error is reported.

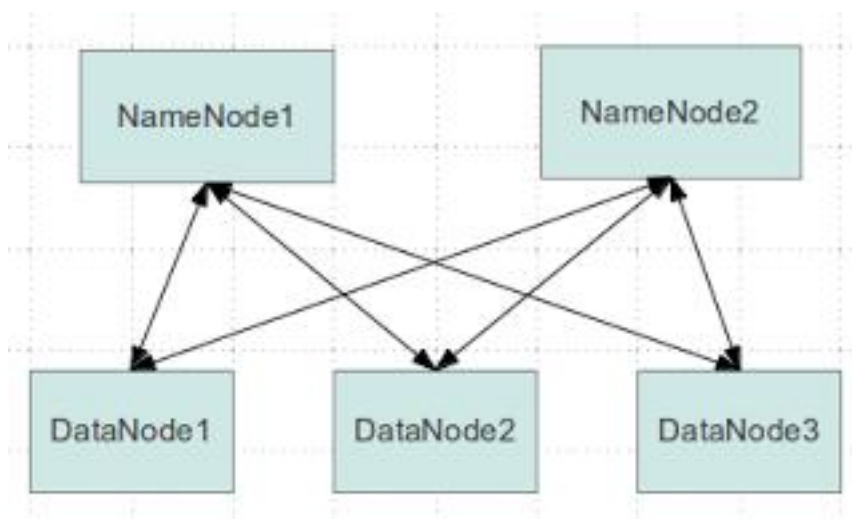
**Answer: C**

**Explanation:**

Note:

\* The current HDFS architecture allows only a single namespace for the entire cluster. A single Namenode manages this namespace. HDFS Federation addresses limitation of current architecture by adding support multiple Namenodes/namespaces to HDFS file system.

\* HDFS Federation enables multiple NameNodes in a cluster for horizontal scalability of NameNode. All these NameNodes work independently and don't require any co-ordination. A DataNode can register with multiple NameNodes in the cluster and can store the data blocks for multiple NameNodes.



**Question No : 18**

Identify two features/issues that MapReduce v2 (MRv2/YARN) is designed to address:

- A. Resource pressure on the JobTracker
- B. HDFS latency.
- C. Ability to run frameworks other than MapReduce, such as MPI.
- D. Reduce complexity of the MapReduce APIs.
- E. Single point of failure in the NameNode.
- F. Standardize on a single MapReduce API.

**Answer: A,C**

**Explanation:** A: MapReduce has undergone a complete overhaul in hadoop-0.23 and we

now have, what we call, MapReduce 2.0 (MRv2) or YARN.

The fundamental idea of MRv2 is to split up the two major functionalities of the JobTracker, resource management and job scheduling/monitoring, into separate daemons. The idea is to have a global ResourceManager (RM) and per-application ApplicationMaster (AM). An application is either a single job in the classical sense of Map-Reduce jobs or a DAG of jobs.

The ResourceManager and per-node slave, the NodeManager (NM), form the data-computation framework. The ResourceManager is the ultimate authority that arbitrates resources among all the applications in the system.

The per-application ApplicationMaster is, in effect, a framework specific library and is tasked with negotiating resources from the ResourceManager and working with the NodeManager(s) to execute and monitor the tasks.

C: YARN, as an aspect of Hadoop, has two major kinds of benefits:

The ability to use programming frameworks other than MapReduce.

Scalability, no matter what programming framework you use.

#### Question No : 19

The most important consideration for slave nodes in a Hadoop cluster running production jobs that require short turnaround times is:

- A. The ratio between the amount of memory and the number of disk drives.
- B. The ratio between the amount of memory and the total storage capacity.
- C. The ratio between the number of processor cores and the amount of memory.
- D. The ratio between the number of processor cores and total storage capacity.
- E. The ratio between the number of processor cores and number of disk drives.

**Answer: E**

#### Question No : 20

The failure of which daemon makes HDFS unavailable on a cluster running MapReduce v1 (MRv1)?

- A. Node Manager
- B. Application Manager
- C. Resource Manager
- D. Secondary NameNode
- E. NameNode
- F. DataNode

**Answer: E**

**Explanation:** The NameNode is the centerpiece of an HDFS file system. It keeps the directory tree of all files in the file system, and tracks where across the cluster the file data is kept. It does not store the data of these files itself. There is only One NameNode process run on any hadoop cluster. NameNode runs on its own JVM process. In a typical production cluster its run on a separate machine. The NameNode is a Single Point of Failure for the HDFS Cluster. When the NameNode goes down, the file system goes offline.

Reference: 24 Interview Questions & Answers for Hadoop MapReduce developers, What is a NameNode? How many instances of NameNode run on a Hadoop Cluster?

#### Question No : 21

Choose three reasons why should you run the HDFS balancer periodically?

- A. To improve data locality for MapReduce tasks.
- B. To ensure that there is consistent disk utilization across the DataNodes.
- C. To ensure that there is capacity in HDFS for additional data.
- D. To ensure that all blocks in the cluster are 128MB in size.
- E. To help HDFS deliver consistent performance under heavy loads.

**Answer: A,B,E**

**Explanation:** The balancer is a tool that balances disk space usage on an HDFS cluster when some datanodes become full or when new empty nodes join the cluster. The tool is deployed as an application program that can be run by the cluster administrator on a live HDFS cluster while applications adding and deleting files.

#### DESCRIPTION

The threshold parameter is a fraction in the range of (0%, 100%) with a default value of 10%. The threshold sets a target for whether the cluster is balanced. A cluster is balanced



if for each datanode, the utilization of the node (ratio of used space at the node to total capacity of the node) differs from the utilization of the (ratio of used space in the cluster to total capacity of the cluster) by no more than the threshold value. The smaller the threshold, the more balanced a cluster will become. It takes more time to run the balancer for small threshold values. Also for a very small threshold the cluster may not be able to reach the balanced state when applications write and delete files concurrently.

The tool moves blocks from highly utilized datanodes to poorly utilized datanodes iteratively. In each iteration a datanode moves or receives no more than the lesser of 10G bytes or the threshold fraction of its capacity. Each iteration runs no more than 20 minutes. At the end of each iteration, the balancer obtains updated datanodes information from the namenode.

Reference: `org.apache.hadoop.hdfs.server.balancer, Class Balancer`

#### Question No : 22

What additional capability does Ganglia provide to monitor a Hadoop?

- A. Ability to monitor the amount of free space on HDFS.
- B. Ability to monitor number of files in HDFS.
- C. Ability to monitor processor utilization.
- D. Ability to monitor free task slots.
- E. Ability to monitor NameNode memory usage.

**Answer: E**

**Explanation:** Ganglia itself collects metrics, such as CPU and memory usage; by using GangliaContext, you can inject Hadoop metrics into Ganglia.

Note:

Ganglia is an open-source, scalable and distributed monitoring system for large clusters. It collects, aggregates and provides time-series views of tens of machine-related metrics such as CPU, memory, storage, network usage.

Ganglia is also a popular solution for monitoring Hadoop and HBase clusters, since Hadoop (and HBase) has built-in support for publishing its metrics to Ganglia. With Ganglia you may easily see the number of bytes written by a particular HDFS datanode over time,

the block cache hit ratio for a given HBase region server, the total number of requests to the HBase cluster, time spent in garbage collection and many, many others.

Hadoop and HBase use GangliaContext class to send the metrics collected by each daemon (such as datanode, tasktracker, jobtracker, HMaster etc) to gmonds.

#### Question No : 23

In a cluster configured with HDFS High Availability (HA) but NOT HDFS federation, each map task run:

- A. In the same Java Virtual Machine as the DataNode.
- B. In the same Java Virtual Machine as the TaskTracker
- C. In its own Java Virtual Machine.
- D. In the same Java Virtual Machine as the JobTracker.

**Answer: C**

**Explanation:** A TaskTracker is a slave node daemon in the cluster that accepts tasks (Map, Reduce and Shuffle operations) from a JobTracker. There is only One Task Tracker process run on any hadoop slave node. Task Tracker runs on its own JVM process. Every TaskTracker is configured with a set of slots, these indicate the number of tasks that it can accept. The TaskTracker starts a separate JVM processes to do the actual work (called as Task Instance) this is to ensure that process failure does not take down the task tracker. The TaskTracker monitors these task instances, capturing the output and exit codes. When the Task instances finish, successfully or not, the task tracker notifies the JobTracker. The TaskTrackers also send out heartbeat messages to the JobTracker, usually every few minutes, to reassure the JobTracker that it is still alive. These message also inform the JobTracker of the number of available slots, so the JobTracker can stay up to date with where in the cluster work can be delegated.

Note: Despite this very high level of reliability, HDFS has always had a well-known single point of failure which impacts HDFS's availability: the system relies on a single Name Node to coordinate access to the file system data. In clusters which are used exclusively for ETL or batch-processing workflows, a brief HDFS outage may not have immediate business impact on an organization; however, in the past few years we have seen HDFS begin to be

used for more interactive workloads or, in the case of HBase, used to directly serve customer requests in real time. In cases such as this, an HDFS outage will immediately impact the productivity of internal users, and perhaps result in downtime visible to external users. For these reasons, adding high availability (HA) to the HDFS Name Node became one of the top priorities for the HDFS community.

Reference: 24 Interview Questions & Answers for Hadoop MapReduce developers , What is a Task Tracker in Hadoop? How many instances of TaskTracker run on a Hadoop Cluster

#### Question No : 24

Where does a MapReduce job store the intermediate data output from Mappers?

- A. On the underlying filesystem of the local disk machine on which the JobTracker ran.
- B. In HDFS, in the job's output directory.
- C. In HDFS, in temporary directory defined mapred.tmp.dir.
- D. On the underlying filesystem of the local disk of the machine on which the Mapper ran.
- E. Stores on the underlying filesystem of the local disk of the machine on which the Reducer.

**Answer: D**

**Explanation:** The mapper output (intermediate data) is stored on the Local file system (NOT HDFS) of each individual mapper nodes. This is typically a temporary directory location which can be setup in config by the hadoop administrator. The intermediate data is cleaned up after the Hadoop Job completes.

Reference: 24 Interview Questions & Answers for Hadoop MapReduce developers , Where is the Mapper Output (intermediate key-value data) stored ?

#### Question No : 25

Which two features does Kerberos security add to a Hadoop cluster?

- A. Authentication for user access to the cluster against a central server
- B. Encryption for data on disk ("at rest")
- C. Encryption on all remote procedure calls (RPCs)
- D. User authentication on all remote procedure calls (RPCs)
- E. Root access to the cluster for users hdfs and mapred but non-root access clients

**Answer: A,D**

**Explanation:**

Hadoop can use the Kerberos protocol to ensure that when someone makes a request, they really are who they say they are. This mechanism is used throughout the cluster. In a secure Hadoop configuration, all of the Hadoop daemons use Kerberos to perform mutual authentication, which means that when two daemons talk to each other, they each make sure that the other daemon is who it says it is. Additionally, this allows the NameNode and JobTracker to ensure that any HDFS or MR requests are being executed with the appropriate authorization level.

Reference: Documentation CDH3 Documentation CDH3 Security Guide, Introduction to Hadoop Security

**Question No : 26**

You are running a Hadoop cluster with NameNode on host mynamenode, a secondary NameNode on host mysecondary and DataNodes.

Which best describes how you determine when the last checkpoint happened?

- A. Execute `hdfs dfsadmin -report` on the command line in and look at the Last Checkpoint information.
- B. Execute `hdfs dfsadmin -saveNameSpace` on the command line which returns to you the last checkpoint value in `fstime` file.
- C. Connect to the web UI of the Secondary NameNode (`http://mysecondarynamenode:50090`) and look at the "Last Checkpoint" information
- D. Connect to the web UI of the NameNode (`http://mynamenode:50070/`) and look at the "Last Checkpoint" information

**Answer: C**

**Explanation:**

Note: SecondaryNameNode: Is the worst name ever given to the module in the history of naming conventions. It is only a check point server which actually gets a back up of the