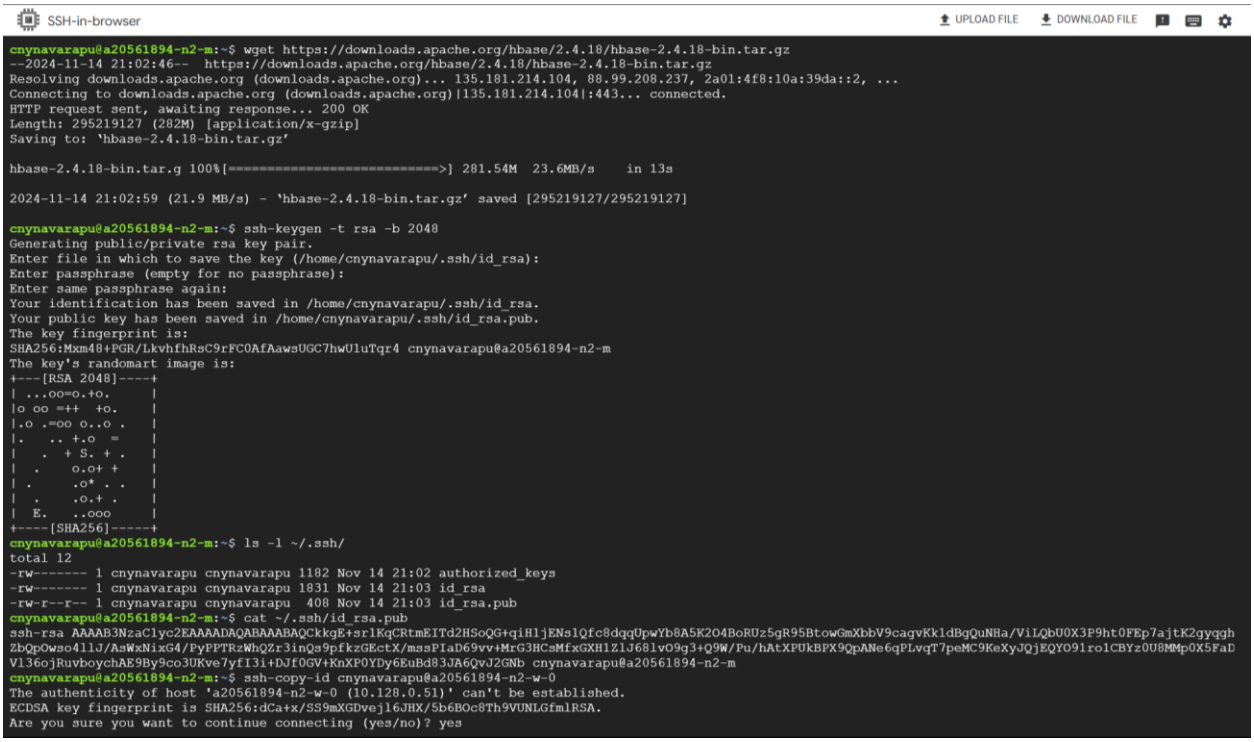
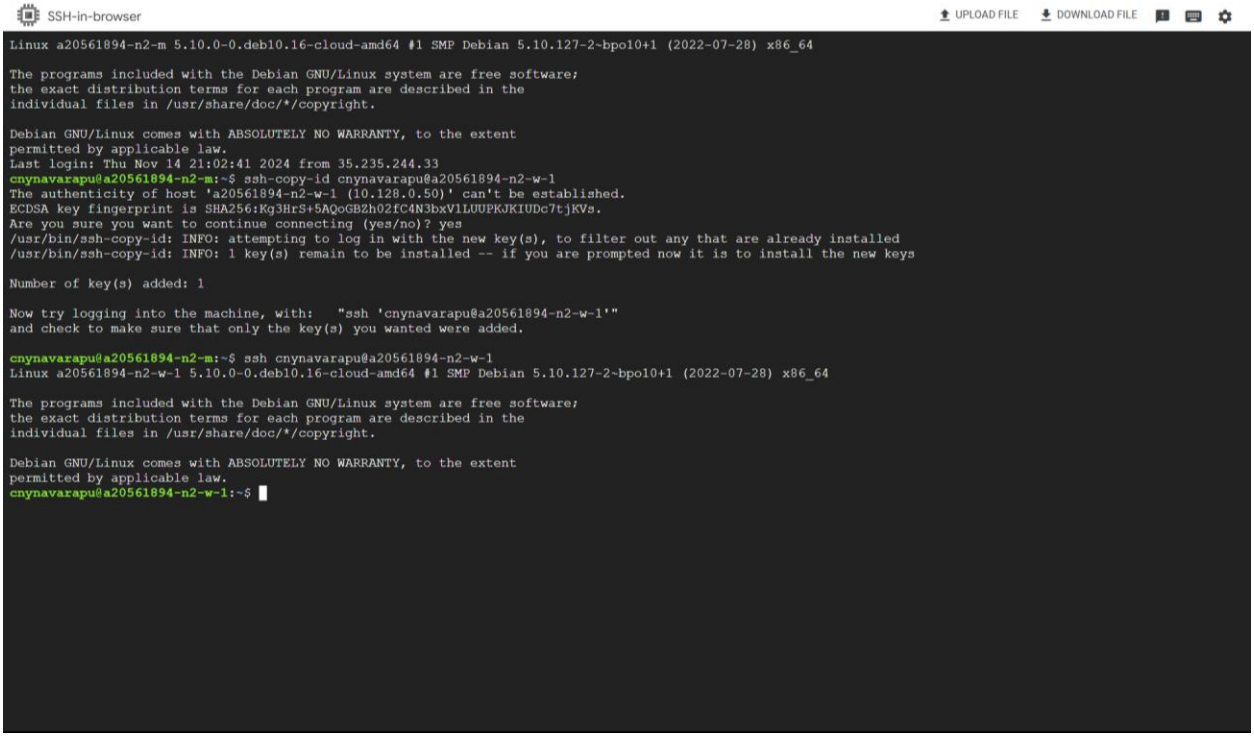
Chaitanya Durgesh Nynavarapu
A20561894

ASSIGNMENT - 10

BIG DATA (CSP 554)





```
Linux a20561894-n2-m 5.10.0-0.deb10.16-cloud-amd64 #1 SMP Debian 5.10.127-2~bpo10+1 (2022-07-28) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Nov 14 21:06:22 2024 from 35.235.245.128
cnynavarapu@a20561894-n2-m:~$ tar -xzf hbase-2.4.18-bin.tar.gz
cnynavarapu@a20561894-n2-m:~$ cd hbase-2.4.18
cnynavarapu@a20561894-n2-m:~/hbase-2.4.18$ vi conf/hbase-site.xml
cnynavarapu@a20561894-n2-m:~/hbase-2.4.18$ vi conf/hbase-site.xml
cnynavarapu@a20561894-n2-m:~/hbase-2.4.18$ nano ~/.bashrc
cnynavarapu@a20561894-n2-m:~/hbase-2.4.18$ source ~/.bashrc
cnynavarapu@a20561894-n2-m:~/hbase-2.4.18$ hbase shell
```

ssh.cloud.google.com/v2/ssh/projects/my-first-project-435417/zones/us-central1-c/instances/a20561894-n2-m?authuser=2&hl=en_US&projectNumber=734966278134&useAdminProxy=true - Google Chrome

ssh.cloud.google.com/v2/ssh/projects/my-first-project-435417/zones/us-central1-c/instances/a20561894-n2-m?authuser=2&hl=en_US&projectNumber=734966278134&useAdminProxy=true

SSH-in-browser UPLOAD FILE DOWNLOAD FILE

```
<configuration>
  <!-- Development/Standalone Configuration -->
  <property>
    <name>hbase.cluster.distributed</name>
    <value>false</value>
  </property>
  <property>
    <name>hbase.tmp.dir</name>
    <value>./tmp</value>
  </property>
  <property>
    <name>hbase.unsafe.stream.capability.enforce</name>
    <value>false</value>
  </property>

  <!-- Cluster Configuration -->
  <property>
    <name>hbase.rootdir</name>
    <value>hdfs://a20561894-n2-m:8020/hbase</value>
  </property>
  <property>
    <name>hbase.cluster.distributed</name>
    <value>true</value>
  </property>
  <property>
    <name>hbase.zookeeper.quorum</name>
    <value>a20561894-n2-m,a20561894-n2-w-1,a20561894-n2-w-0</value>
  </property>
</configuration>
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                                    29,17         All
```

Exercise 1:

```
Took 0.0018 seconds
hbase:001:0> create 'csp554Tbl', 'cf1', 'cf2'
Created table csp554Tbl
Took 2.2954 seconds
=> Hbase::Table - csp554Tbl
hbase:002:0> describe 'csp554Tbl'
Table csp554Tbl is ENABLED
csp554Tbl
COLUMN FAMILIES DESCRIPTION
{NAME => 'cf1', BLOOMFILTER => 'ROW', IN_MEMORY => 'false', VERSIONS => '1', KEEP_DELETED_CELLS => 'FALSE', DATA_BLOCK_ENCODING => 'NONE', COMPRESSION => 'NONE',
 TTL => 'FOREVER', MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0'}

{NAME => 'cf2', BLOOMFILTER => 'ROW', IN_MEMORY => 'false', VERSIONS => '1', KEEP_DELETED_CELLS => 'FALSE', DATA_BLOCK_ENCODING => 'NONE', COMPRESSION => 'NONE',
 TTL => 'FOREVER', MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0'}

2 row(s)
Quota is disabled
```

Exercise 2:

```
Took 0.1342 seconds
hbase:003:0> put 'csp554Tbl', 'Row1', 'cf1:name', 'Sam'
Took 0.0902 seconds
hbase:004:0> put 'csp554Tbl', 'Row2', 'cf1:name', 'Ahmed'
Took 0.0072 seconds
hbase:005:0> put 'csp554Tbl', 'Row1', 'cf2:job', 'Pilot'
Took 0.0090 seconds
hbase:006:0> put 'csp554Tbl', 'Row2', 'cf2:job', 'Doctor'
Took 0.0053 seconds
hbase:007:0> put 'csp554Tbl', 'Row1', 'cf2:level', 'LZ3'
Took 0.0060 seconds
hbase:008:0> put 'csp554Tbl', 'Row2', 'cf2:level', 'AR7'
Took 0.0053 seconds
```

```
hbase:009:0> scan 'csp554Tbl'
ROW                                   COLUMN+CELL
 Row1                                 column=cf1:name, timestamp=2024-11-14T18:30:19.364, value=Sam
 Row1                                 column=cf2:job, timestamp=2024-11-14T18:30:19.421, value=Pilot
 Row1                                 column=cf2:level, timestamp=2024-11-14T18:30:19.490, value=LZ3
 Row2                                 column=cf1:name, timestamp=2024-11-14T18:30:19.394, value=Ahmed
 Row2                                 column=cf2:job, timestamp=2024-11-14T18:30:19.463, value=Doctor
 Row2                                 column=cf2:level, timestamp=2024-11-14T18:30:28.328, value=AR7
2 row(s)
Took 0.0356 seconds
```

Exercise 3:

```
hbase:010:0> get 'csp554Tbl', 'Row1', 'cf2:level'
COLUMN                                CELL
 cf2:level                            timestamp=2024-11-14T18:30:19.490, value=LZ3
1 row(s)
Took 0.0134 seconds
```

Exercise 4:

```
hbase:011:0> get 'csp554Tbl', 'Row2', 'cf1:name'
COLUMN                                CELL
 cf1:name                             timestamp=2024-11-14T18:30:19.394, value=Ahmed
1 row(s)
Took 0.0076 seconds
```

Exercise 5:

```
hbase:012:0> scan 'csp554Tbl', {LIMIT => 2}
ROW                                   COLUMN+CELL
 Row1                                 column=cf1:name, timestamp=2024-11-14T18:30:19.364, value=Sam
 Row1                                 column=cf2:job, timestamp=2024-11-14T18:30:19.421, value=Pilot
 Row1                                 column=cf2:level, timestamp=2024-11-14T18:30:19.490, value=LZ3
 Row2                                 column=cf1:name, timestamp=2024-11-14T18:30:19.394, value=Ahmed
 Row2                                 column=cf2:job, timestamp=2024-11-14T18:30:19.463, value=Doctor
 Row2                                 column=cf2:level, timestamp=2024-11-14T18:30:28.328, value=AR7
2 row(s)
Took 0.0158 seconds
```

Exercise 6:

A Novel HBase Data Storage in Wireless Sensor Networks

Introduction to the Problem

Wireless sensor networks (WSNs) generate enormous amounts of data that traditional storage systems often struggle to manage. These networks rely on distributed sensors, which gather data continuously across various regions. Managing this high-speed, high-volume data efficiently is critical, especially in real-time applications like environmental monitoring and smart cities. This study addresses the need for a storage solution that can handle WSN data's unique demands.

Purpose and Hypothesis

The article explores the potential of using an optimized HBase-based storage system to improve data management in WSNs. The researchers hypothesize that a strategically configured HBase cluster could meet WSNs' real-time processing requirements, balancing quick data access, efficient storage usage, and seamless retrieval.

Methodology

- **Two-layer Distributed Storage Model**
  - To structure the data effectively, the researchers developed a two-layer distributed model in HBase. They used Extenics primitives—a formalization approach to organize heterogeneous datasets—allowing seamless storage and retrieval of data from multiple sources across different regions.
- **Multi-threaded Data Buffering and Partitioning**
  - A multi-threaded approach was adopted for data buffering and partitioning, enabling faster data processing. This system efficiently organizes data before storage, reducing bottlenecks caused by high-speed data flow.
- **Filters and Coprocessors**
  - To enhance data retrieval, HBase's filtering and coprocessing features were utilized. These tools enable the system to handle large data sets and distribute complex processing tasks across HBase servers, which improves retrieval speed and efficiency.
- **Dynamic Node Updating for Scalability**
  - A dynamic updating mechanism for HBase nodes allows the system to expand as data demands grow. This feature helps ensure the storage solution remains scalable and adaptable to increasing data volumes.

Results and Findings

- **Performance Gains in Speed and Efficiency**
  - The optimized HBase storage system demonstrated notable improvements in both speed and efficiency over single-node storage configurations. Multi-threaded data processing significantly reduced delays and boosted throughput, while the cluster-based model achieved faster write and read times, proving that the HBase architecture is effective for real-time data processing.
- **Specific Performance Metrics**
  - The article details a comprehensive performance analysis, including tests that compare write and read times, delays at different buffer thresholds, and a comparison to Oracle's streaming data storage. The system's performance was shown to surpass traditional storage setups, particularly in high-speed data scenarios typical of WSNs.

Discussion and Future Directions

System Suitability and Potential Improvements

The findings suggest that an optimized HBase cluster is highly suitable for WSN data management, especially in applications demanding real-time data access and large-scale data integration. The authors also discuss potential future improvements, such as enhancing fault tolerance and optimizing storage space management to further boost reliability.

Conclusion

The study provides a compelling case for using HBase as a storage solution in WSNs. With the right optimizations, HBase can handle the specific demands of WSNs, including real-time data processing and scalability. This research contributes to the field of WSN data management, offering practical insights and solutions for data-intensive applications.