

Ex. No: 1	Linux Commands
16th July, 2024	

Aim:

To find and implement basic Linux Commands.

Programs and Outputs:

1. Display information about files in the current directory.

```
[→ Desktop ls -l
total 12537696
-rw-r--r--@ 1 daver staff      40286 Jul 14 14:39 0087502563d4331bea33323f19f
c8d1a.jpg
drwxr-xr-x  6 daver staff      192 Jul 16 11:27 ACM Summer School
drwxr-xr-x  15 daver staff     480 Jun 11 09:40 Alpha Course
drwxr-xr-x@ 88 daver staff    2816 Jul  9 15:24 Compiled
drwxr-xr-x@  5 daver staff     160 Mar 25 17:49 Daddy Friends Chats
-rw-r--r--@  1 daver staff   2622196 Jul 15 17:16 Essentialism.pdf
drwxr-xr-x  33 daver staff    1056 Jul 13 07:46 Masters
-rw-r--r--@  1 daver staff   24194136 Jun  5 06:57 My Chats till 5th June 2024
.zip
-rw-r--r--@  1 daver staff      405 Jun 28 20:01 New Phone Password.rtf
-rw-r--r--  1 daver staff  6387186126 Jul 10 13:53 PLM Backup - 10th July 2024
.zip
-rw-r--r--@  1 daver staff   572210 Jul 16 13:23 Screenshot 2024-07-16 at 1.
23.29 PM.png]
```

14. Search for a specific string in an output
15. Display active processes on the terminal

```
[→ ~ grep "awesome" Hello.txt
This is awesome
→ ~ ]
```

```
[→ ~ top
```

Desktop — top — top — top — One Dark — 80x24

```

Processes: 623 total, 5 running, 1 stuck, 617 sleeping, 2846 threads 13:41:28
Load Avg: 1.35, 1.52, 1.68 CPU usage: 5.52% user, 3.21% sys, 91.26% idle
SharedLibs: 787M resident, 129M data, 60M linkedit.
MemRegions: 139817 total, 4382M resident, 442M private, 2180M shared.
PhysMem: 15G used (1933M wired, 1474M compressor), 223M unused.
VM: 243T vsize, 4921M framework vsize, 28(0) swapins, 92(0) swapouts.
Networks: packets: 23007953/22G in, 11987591/4272M out.
Disks: 15593238/305G read, 11596664/196G written.

PID COMMAND %CPU TIME #TH #WQ #PORT MEM PURG CMPRS PGRP
366 WindowServer 27.2 11:32:39 22/1 6 5433+ 702M- 141M- 237M 366
911 Google Drive 7.6 02:44:07 11/1 3/1 159 46M 0B 29M 843
433 coreaudiod 4.9 73:10.95 15 8 1987+ 36M+ 0B 11M 433
915 Google Drive 4.7 01:42:58 18/2 1 347 70M 0B 59M 843
0 kernel_task 4.6 04:27:01 565/10 0 0 70M- 0B 0B 0
55180 top 4.2 00:00.43 1/1 0 31+ 6449K+ 0B 0B 55180
55071 Terminal 4.0 00:01.44 8 3 281- 49M+ 9200K 0B 55071
370 runningboard 2.2 31:40.92 8 7 803+ 12M 0B 1024K 370
303 mediaremoted 1.9 01:29.99 11 8 288+ 7617K+ 0B 1696K 303
54429 com.apple.We 1.7 00:25.29 8 4 85 182M 0B 0B 54429
368 tccd 1.6 01:46.71 5 4 59+ 6945K+ 32K 1456K 368
9189 CharacterPal 1.4 04:14.97 3 1 324- 78M- 0B 79M 9189
51364 Safari 1.2 02:44.83 14 5 1218+ 157M 121M 40M 51364
54419 Obsidian Hel 0.9 00:41.82 35 1 323 242M+ 0B 0B 54419

```

16. Download files from the internet. (wget needs to be installed)
17. Create or update passwords for existing user

```
[→ Desktop curl -O "https://heream.wordpress.com/wp-content/uploads/2017/12/greg-mckeown-essentialism-the-disciplined-pursuit.pdf"
% Total    % Received % Xferd  Average Speed   Time     Time     Time  Current
          Dload  Upload Total   Spent    Left  Speed
100 1790k  100 1790k    0      0  283k      0  0:00:06  0:00:06 --:--:--  415k
→ Desktop ]
```

```
○ ○ ○ daver — sudo dscl . -passwd /Users/daver 12345 — dscl — dscl ↵ sudo — One Dark — 8...
[→ Desktop cd ..
[→ ~ sudo dscl . -passwd /Users/daver 12345
[Password:
Permission denied. Please enter user's old password:
```

18. View the exact location of any tool/software installed

```
[→ ~ which which  
which: shell built-in command  
[→ ~ which ls  
ls: aliased to ls -G  
→ ~
```

19. Check the details of the file system

```
[→ ~ df -h
Filesystem      Size   Used  Avail Capacity iused ifree %iused Mounted on
/dev/disk3s3s1  460Gi  9.6Gi  230Gi    4%    404k  2.4G  0%   /
devfs          208Ki  208Ki   0Bi  100%    720   0  100%  /dev
/dev/disk3s6   460Gi  1.0Gi  230Gi    1%     1  2.4G  0%  /System/Volu
mes/VM
/dev/disk3s4   460Gi  5.7Gi  230Gi    3%    1.1k  2.4G  0%  /System/Volu
mes/Preboot
/dev/disk3s2   460Gi  52Mi   230Gi    1%     52  2.4G  0%  /System/Volu
mes/Update
/dev/disk1s2   500Mi  6.0Mi  480Mi    2%     1  4.9M  0%  /System/Volu
mes/xarts
/dev/disk1s1   500Mi  6.2Mi  480Mi    2%     32  4.9M  0%  /System/Volu
mes/iSCPPreboot
/dev/disk1s3   500Mi  2.8Mi  480Mi    1%     64  4.9M  0%  /System/Volu
mes/Hardware
/dev/disk3s1   460Gi  213Gi  230Gi   49%    1.5M  2.4G  0%  /System/Volu
mes/Data
map auto_home   0Bi    0Bi    0Bi  100%     0    0   -  /System/Volu
→ ~ ]
```

20. Check the lines, word count, and characters in a file using different options

```
[→ Desktop wc -l Hello.txt
      2 Hello.txt
[→ Desktop wc -w Hello.txt
      13 Hello.txt
[→ Desktop wc -c Hello.txt
      59 Hello.txt
→ Desktop ]
```

Result:

Successfully implemented and Tested Linux Commands.

Ex. No: 2	Install Hadoop
23rd July, 2024	

Aim:

To install Hadoop in Ubuntu.

Program:

Steps from this link : <https://medium.com/@abhikdey06/apache-hadoop-3-3-6-installation-on-ubuntu-22-04-14516bceec85>

1. Install Java:

- Check if Java is installed: `java -version`
- If Java is not installed, download and install it:

```

```
sudo apt update
sudo apt install openjdk-8-jdk
```

```

2. Download Hadoop:

- Go to the Apache Hadoop releases page and download the binary distribution (e.g., Hadoop 3.4.0):

```
wget https://downloads.apache.org/hadoop/common/hadoop-3.4.0/hadoop-3.4.0.tar.gz
```

- Extract the downloaded file:

```
tar -xzvf hadoop-3.4.0.tar.gz
```

3. Move Hadoop to the desired directory: `sudo mv hadoop-3.4.0 /opt/hadoop`

4. Set environment variables:

- Open the `~/.bashrc` file for editing: `nano ~/.bashrc`
- Add the following lines at the end of the file:

```
export HADOOP_HOME=/opt/hadoop  
export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
```

- Save and exit, then refresh your terminal: `source ~/.bashrc`

5. Configure Hadoop:

- Navigate to the Hadoop configuration directory: `cd $HADOOP_HOME/etc/hadoop/`
- Edit `hadoop-env.sh` to set the Java home path: `nano hadoop-env.sh`
- Add or modify the line to include your Java installation path: `export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64`

6. Format the NameNode: `hadoop namenode -format`

7. Start YARN: `start-yarn.sh`

8. Start all Hadoop services: `start-all.sh`

9. Check running processes: `jps`

10. Access Hadoop web interface: `http://localhost:9870`

11. Stop all Hadoop services when needed: `stop-all.sh`

Output:

Result:

Successfully Installed Hadoop on Ubuntu Linux.

Ex. No: 3	
30th July, 2024	

MapReduce in Hadoop

Aim:

To implement MapReduce for the word count problem in Hadoop.

Program:

```

import java.io.IOException;
import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class WordCount {

    public static class TokenizerMapper
        extends Mapper<Object, Text, Text, IntWritable>{

        private final static IntWritable one = new IntWritable(1);
        private Text word = new Text();

        public void map(Object key, Text value, Context context
                       ) throws IOException, InterruptedException {
            StringTokenizer itr = new StringTokenizer(value.toString());
            while (itr.hasMoreTokens()) {
                word.set(itr.nextToken());
                context.write(word, one);
            }
        }
    }

    public static class IntSumReducer
        extends Reducer<Text,IntWritable,Text,IntWritable> {
        private IntWritable result = new IntWritable();

        public void reduce(Text key, Iterable<IntWritable> values,
                           Context context
                           ) throws IOException, InterruptedException {
            int sum = 0;
            for (IntWritable val : values) {
                sum += val.get();
            }
            result.set(sum);
        }
    }
}

```

```

        context.write(key, result);
    }
}

public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "word count");
    job.setJarByClass(WordCount.class);
    job.setMapperClass(TokenizerMapper.class);
    job.setCombinerClass(IntSumReducer.class);
    job.setReducerClass(IntSumReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
}
}

```

Program (Python):

Mapper.py

```
#!/usr/bin/env python3
```

```
import sys
```

```

# Input comes from standard input (stdin)
for line in sys.stdin:
    # Remove leading and trailing whitespace
    line = line.strip()
    # Split the line into words
    words = line.split()
    # Increase counters
    for word in words:
        # Write the results to standard output (stdout)
        # What we output here will be the input for the Reducer
        # The tab-delimited format is <word, 1>
        print('%s\t%s' % (word, 1))

```

reducer.py

```
#!/usr/bin/env python3

import sys

current_word = None
current_count = 0
word = None

# Input comes from standard input (stdin)
for line in sys.stdin:
    # Remove leading and trailing whitespace
    line = line.strip()
    # Parse the input we got from mapper.py
    word, count = line.split('\t', 1)
    # Convert count (currently a string) to int
    try:
        count = int(count)
    except ValueError:
        # Count was not a number, so ignore/discard this line
        continue
    # This IF-switch only works because Hadoop sorts map output
    # by key (here: word) before it is passed to the reducer
    if current_word == word:
        current_count += count
    else:
        if current_word:
            # Write result to standard output (stdout)
            print('%s\t%s' % (current_word, current_count))
        current_count = count
        current_word = word
```

```
# Do not forget to output the last word if needed!
if current_word == word:
    print('%s\t%s' % (current_word, current_count))
```

commands :

```
chmod +x mapper.py
chmod +x reducer.py
```

```
$HADOOP_HOME/bin/hadoop jar $HADOOP_HOME/hadoop-streaming.jar \
-input myInputDirs \
-output myOutputDir \
-mapper /bin/cat \
-reducer /usr/bin/wc
```

```
mapred streaming -input /wordcount/input -output /wordcount/output4 -mapper
./mapper.py -reducer ./reducer.py
```

Output:

```
hadoop@DESKTOP-QDJVLB4:~/program_files$ hadoop fs -cat /wordcount/output4/part-00000
1926, 1
21, 1
Albert, 1
Alexandra 1
April 1
As 1
Auxiliary 1
Born 1
Bowes-Lyon. 1
Buckingham 1
Castle. 1
Despite 1
Duke 1
Edward 1
Elizabeth 7
Elizabeth's 1
George 1
Her 3
```

Result:

Successfully implemented a MapReduce Function using Hadoop.

Ex. No: 4	MapReduce Problems
6th August, 2024	

Aim:

1. To Implement map reduce for NCDC weather dataset using Hadoop and find the max and min temperature.
2. Implement Apriori algorithm using map reduce paradigm.

Program 1:

TemperatureMapper.java

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.FloatWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;

public class TemperatureMapper extends Mapper<LongWritable, Text, Text, FloatWritable> {

    private final static FloatWritable temperature = new FloatWritable();
    private final Text tmaxKey = new Text("TMAX");
    private final Text tminKey = new Text("TMIN");

    @Override
    protected void map(LongWritable key, Text value, Context context)
        throws IOException, InterruptedException {
        String line = value.toString();
        // Split the line on commas for CSV file
        String[] fields = line.split(",");

```

```
// Skip header row and incomplete rows
if (fields[0].equals("STATION") || fields.length < 6) {
    return;
}

// Safely parse TMAX and TMIN values
try {
    String tmaxStr = fields[4].trim();
    tmaxStr = tmaxStr.replace("'", ' ');
    String tminStr = fields[5].trim();
    tminStr = tminStr.replace("'", ' ');
    // System.out.println(tmaxStr);

    if (!tmaxStr.isEmpty() && !tminStr.isEmpty()) {
        float tmax = Float.parseFloat(tmaxStr);
        float tmin = Float.parseFloat(tminStr);

        temperature.set(tmax);
        context.write(tmaxKey, temperature);
        temperature.set(tmin);
        context.write(tminKey, temperature);
    }
} catch (NumberFormatException e) {
    // Log parsing errors for debug purposes
    System.err.println("Error parsing temperature: " +
e.getMessage());
}
}
```

TemperatureReducer.java

```
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.FloatWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

import java.io.IOException;

public class TemperatureReducer extends Reducer<Text, FloatWritable,
Text, FloatWritable> {

    @Override
    protected void reduce(Text key, Iterable<FloatWritable> values,
Context context) throws IOException, InterruptedException {
        Float extremeValue;

        if (key.toString().equals("TMAX")) {
            extremeValue = Float.MIN_VALUE;
            for (FloatWritable value : values) {
                extremeValue = Math.max(extremeValue, value.get());
            }
            System.out.println("Maximum Temperature: " + extremeValue);
        } else { // TMIN
            extremeValue = Float.MAX_VALUE;
            for (FloatWritable value : values) {
                extremeValue = Math.min(extremeValue, value.get());
            }
            System.out.println("Minimum Temperature: " + extremeValue);
        }

        // Emit the result
    }
}
```

```
        context.write(key, new FloatWritable(extremeValue));  
    }  
}
```

TemperatureDriver.java

```
import org.apache.hadoop.conf.Configuration;  
import org.apache.hadoop.fs.Path;  
import org.apache.hadoop.io.FloatWritable;  
import org.apache.hadoop.io.Text;  
import org.apache.hadoop.mapreduce.Job;  
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;  
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;  
  
public class TemperatureDriver {  
    public static void main(String[] args) throws Exception {  
        Configuration conf = new Configuration();  
        Job job = Job.getInstance(conf, "Max and Min Temperature");  
  
        job.setJarByClass(TemperatureDriver.class);  
        job.setMapperClass(TemperatureMapper.class);  
        job.setReducerClass(TemperatureReducer.class);  
  
        job.setOutputKeyClass(Text.class);  
        job.setOutputValueClass(FloatWritable.class);  
  
        FileInputFormat.addInputPath(job, new Path(args[0]));  
        FileOutputFormat.setOutputPath(job, new Path(args[1]));  
  
        System.exit(job.waitForCompletion(true) ? 0 : 1);  
    }  
}
```

```
}
```

Program 2 :

Mapper.py

```
#!/usr/bin/env python
import sys
from itertools import combinations

def generate_combinations(item_list, length):
    return list(combinations(item_list, length))

# Input comes from standard input (stdin)
for line in sys.stdin:
    line = line.strip()
    items = line.split()
    for length in range(1, len(items) + 1):
        for combination in generate_combinations(items, length):
            print(f"{''.join(combination)}\t1")
```

Reducer.py

```
#!/usr/bin/env python
import sys

current_itemset = None
current_count = 0

# Input comes from standard input (stdin)
for line in sys.stdin:
    line = line.strip()
```

```

itemset, count = line.split('\t', 1)
count = int(count)

if current_itemset == itemset:
    current_count += count
else:
    if current_itemset:
        print(f"{current_itemset}\t{current_count}")
    current_count = count
    current_itemset = itemset

if current_itemset == itemset:
    print(f"{current_itemset}\t{current_count}")

```

Outputs :

The screenshot displays two windows side-by-side. On the left is a terminal window titled 'hadoop@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~/hadoop'. It contains the following text:

```

root@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC: /home/hadoop -> hadoop@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~/hadoop$ hdfs dfs -cat MyProject/output/part-00000
The minimum temperature is: -36.3
The maximum temperature is: 9.9
hadoop@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/hadoop$ 

```

On the right is a web browser window titled 'Browsing HDFS' with the URL 'localhost:9870/explorer.html#/user/hadoop/MyProject'. The page title is 'Hadoop' and it shows a navigation bar with 'Overview', 'Datanodes', 'Datanode Volume Failures', 'Snapshot', and 'Startup Progress'. Below the navigation bar is a 'Utilities' dropdown menu.

Below the browser window is a 'Browse Directory' section. It has a search bar with '/user/hadoop/MyProject/output' and a 'Go!' button. There are also icons for refresh, search, and file operations. A table below shows the contents of the directory:

	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
<input type="checkbox"/>	-rw-r--r--	hadoop	supergroup	66 B	Sep 09 10:57	1	128 MB	part-00000

At the bottom of the table, it says 'Showing 1 to 1 of 1 entries' and has 'Previous' and 'Next' buttons. A note at the bottom of the page says 'Hadoop, 2023.'

hadoop@snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~/hadoop

```
root@snuce-HP-Pro-Tower-400-G9-PCI-Desktop-PC: /home/h... x hadoop@snuce-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~/had... x
hadoop@snuce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/hadoop$ hdfs dfs -cat MyProject/output/part-00000
The minimum temperature is: -36.3
The maximum temperature is: 9.9
hadoop@snuce-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/hadoop$
```

10.23.22.52:9870/explorer.html#/MyOutput

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities ▾

Browse Directory

/MyOutput Go!

Show 25 entries Search:

	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	-rW-r--r--	hadoop	supergroup	0 B	Sep 05 13:09	1	128 MB	_SUCCESS	
<input type="checkbox"/>	-rw-r--r--	hadoop	supergroup	7.91 KB	Sep 05 13:09	1	128 MB	part-r-00000	

Showing 1 to 2 of 2 entries Previous 1 Next

Hadoop, 2023.

10.23.22.52:9870/explorer.html#/MyOutput

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

File information - part-r-00000

Download Head the file (first 32K) Tail the file (last 32K)

Block information - Block 0

Block ID: 1073741826
Block Pool ID: BP-1383049739-127.0.1-1725521904815
Generation Stamp: 1002
Size: 8104
Availability:
• snucse-HP-Pro-Tower-400-G9-PCI-Desktop-PC

Block Size Name MB _SUCCESS MB part-r-00000

Search:

Previous 1 Next

Browse Directory /MyOutput

Show 25 entries

Permission Owner

-r--r--r-- hadoop

-r--r--r-- hadoop

Showing 1 to 2 of 2 entries

Hadoop, 2023.

Home / Downloads

Recent Starred Home Desktop Documents Downloads Music Pictures Videos Trash Other Locations

part-r-00000 hadoop-mapreduce-client-co... hadoop-common-3.3.6.jar CRND0103-2021-AK-Denali_2...

Result:

Successfully implemented a Weather Analysis and Apriori Algorithm using MapReduce

Ex. No: 5	Apache Spark
13th August, 2024	

Aim:

To install Spark and PySpark and run the WordCount Program and Distribution of Ratings in MovieLens Dataset.

Program:

1. Download and Install Spark
 - a. wget <https://www.apache.org/dyn/closer.lua/spark/spark-3.5.0/spark-3.5.0-bin-hadoop3.tgz>
 - b. tar -xzf spark-3.5.0-bin-hadoop3.tgz
 - c. mv spark-3.5.0-bin-hadoop3 spark
2. Set Environment Variables
 - a. nano ~/.bashrc
 - b. export SPARK_HOME=~/spark
 - c. export PATH=\$PATH:\$SPARK_HOME/bin
 - d. export PYTHONPATH=\$SPARK_HOME/python:\$PYTHONPATH
 - e. export PYSPARK_PYTHON=python3
 - f. source ~/.bashrc
3. Running Programs : spark-submit file.py

WordCount Program

```
import findspark
findspark.init()

from pyspark.sql import SparkSession
from pyspark.conf import SparkConf

spark =
SparkSession.builder.master("local[*]").appName("WordCount").getOrCreate()
```

```
sc = spark.sparkContext

text = "./words.txt"

text_read = sc.textFile(text)

text_read.take(20)

words = text_read.flatMap(lambda x: x.split())

words.take(15)

words = words.map(lambda word: word.rstrip('.'))

words.take(15)

word_count = words.map(lambda word: (word, 1))
word_count.take(20)

word_count_red = word_count.reduceByKey(lambda x, y: (x+y)).sortByKey()

word_count_red = word_count_red.map(lambda x: (x[1], x[0]))

word_count_red.take(15)
```

MovieLens Dataset

```
import findspark
findspark.init()
```

```
from pyspark.sql import SparkSession
from pyspark.conf import SparkConf

spark =
SparkSession.builder.master("local[*]").appName("MovieLens").getOrCreate()

sc = spark.sparkContext

text = "./u.data"

text_read = spark.read.csv(text)

import pyspark.pandas as ps

data = ps.read_csv(text, sep="\t", names = ["userid", "movieid",
"rating", "idid"])

data

len(data.groupby(['userid']).describe()['rating'])
```

Output:

	userid	movieid	rating	idid
0	196	242	3	881250949
1	186	302	3	891717742
2	22	377	1	878887116
3	244	51	2	880606923
4	166	346	1	886397596
5	298	474	4	884182806
6	115	265	2	881171488
7	253	465	5	891628467
8	305	451	3	886324817
9	6	86	3	883603013
10	62	257	2	879372434
11	286	1014	5	879781125
12	200	222	5	876042340
13	210	40	3	891035994
14	224	29	3	888104457
15	303	785	3	879485318
16	122	387	5	879270459

Result:

Successfully Installed Spark and PySpark, and implemented the WordCount Program and Distribution of Movie Reviews using them.

Ex. No: 6	PySpark 2
20th August, 2024	

Aim:

- (a) To use the “friends _ test” dataset. Col1 is ID, Col2 is name, Col 3 is Age, Col 4 is num of friends. Understand mapvalues function of RDD in spark and find the average number of friends for each unique age present in the dataset.
- (b) Use the “temp.csv” dataset. Column headers are present in the dataset. Understand filter operations and filter out only the “TMIN” values from the “desc” column. With the resultant data (RDD) find the following:
 - a. Minimum temperature (overall)
 - b. Minimum temperature for every ItemID
 - c. Minimum temperature for every StationID.
- (c) Use the same dataset, filter only “TMAX” column and find the maximum temperatures just like the ones mentioned above.

Program:

```

from pyspark import SparkConf, SparkContext
from pyspark.sql import *
from pyspark.sql.functions import *
from pyspark.sql.types import *

spark = SparkSession.builder.appName("friends
test").config("spark.memory.offHeap.e
df = spark.read.csv('friends_test.csv',header=False)
df.explain()
spark.stop()
conf = SparkConf().setAppName("Basicapp").setMaster("local[*]")
sc = SparkContext(conf=conf)
rdd = sc.textFile("friends_test.csv")
rdd.first()
rdd_split = rdd.map(lambda line: line.split(","))
for row in rdd_split.take(5):
    print(row)
  
```

RDD approach for average number of friends

```
age_friends_rdd = rdd_split.map(lambda row: (int(row[2]), (int(row[3]), 1)))
sum_count_rdd = age_friends_rdd.reduceByKey(lambda a, b: (a[0] + b[0], a[1] + b[1]))
avg_friends_by_age = sum_count_rdd.mapValues(lambda x: x[0] / x[1])
sorted_avg_friends_by_age = avg_friends_by_age.sortByKey()
for age, avg_friends in sorted_avg_friends_by_age.collect():
    print(f"Age: {age}, Average Number of Friends: {avg_friends:.2f}")
age_friends_rdd = rdd_split.map(lambda row: (int(row[2]), int(row[3])))
max_friends_by_age = age_friends_rdd.reduceByKey(lambda a, b: a if a > b else b)
min_friends_by_age = age_friends_rdd.reduceByKey(lambda a, b: a if a < b else b)
max_min_friends_by_age = max_friends_by_age.join(min_friends_by_age)
sorted_max_min_friends_by_age = max_min_friends_by_age.sortByKey()
for age, (max_friends, min_friends) in sorted_max_min_friends_by_age.collect():
    print(f"Age: {age}, Max Friends: {max_friends}, Min Friends: {min_friends}")
sc.stop()
```

Temp dataset

```
conf = SparkConf().setAppName("temp_dataset").setMaster("local[*]")
sc = SparkContext(conf=conf)
rdd = sc.textFile("temp.csv")
rdd.first()
rdd_header = rdd.first()
rdd_filter = rdd.filter(lambda row: row != rdd_header)
rdd_data = rdd_filter.map(lambda row: row.split(","))
def rdd_display(x, threshold=5):
    count = 0
    for i in x.collect():
        print(i)
```

```

        count+=1
        if(count>threshold):
            break
    rdd_display(rdd_data)
    rdd_TMIN_filter = rdd_data.filter(lambda row:row[2]=="TMIN")
    rdd_display(rdd_TMIN_filter)

    rdd_min_overall = rdd_TMIN_filter.map(lambda x:int(x[3])).reduce(lambda
a,b:a if a
print("Minimum temperature overall",rdd_min_overall)
    rdd_min_itemID = rdd_TMIN_filter.map(lambda
x:(x[0],int(x[3]))).reduceByKey(lambda
print("minimum temperature by itemID")
    rdd_display(rdd_min_itemID)
    rdd_min_stationID = rdd_TMIN_filter.map(lambda
x:(x[1],int(x[3]))).reduceByKey(lambda
print("minimum temperature by StationID")
    rdd_display(rdd_min_stationID,10)
    rdd_TMAX_filter = rdd_data.filter(lambda row:row[2]=="TMAX")
    rdd_display(rdd_TMAX_filter)
    rdd_max_overall = rdd_TMAX_filter.map(lambda x:int(x[3])).reduce(lambda
a,b:a if a
print("Maximum temperature overall",rdd_max_overall)
    rdd_max_itemID = rdd_TMAX_filter.map(lambda
x:(x[0],int(x[3]))).reduceByKey(lambda
print("maximum temperature by itemID")
    rdd_display(rdd_max_itemID)
    rdd_max_stationID = rdd_TMAX_filter.map(lambda
x:(x[1],int(x[3]))).reduceByKey(lambda
print("maximum temperature by StationID")
    rdd_display(rdd_max_stationID,10)
sc.stop()

```

Output:

Age: 18, Average Number of Friends: 343.38
Age: 19, Average Number of Friends: 213.27
Age: 20, Average Number of Friends: 165.00
Age: 21, Average Number of Friends: 350.88
Age: 22, Average Number of Friends: 286.43
Age: 23, Average Number of Friends: 246.30
Age: 24, Average Number of Friends: 233.80
Age: 25, Average Number of Friends: 197.45
Age: 26, Average Number of Friends: 242.06
Age: 27, Average Number of Friends: 228.12
Age: 28, Average Number of Friends: 209.10
Age: 29, Average Number of Friends: 215.92
Age: 30, Average Number of Friends: 235.82
Age: 31, Average Number of Friends: 267.25
Age: 32, Average Number of Friends: 207.91
Age: 33, Average Number of Friends: 325.33
Age: 34, Average Number of Friends: 245.50
Age: 35, Average Number of Friends: 211.62
Age: 36, Average Number of Friends: 246.60
Age: 37, Average Number of Friends: 249.33
Age: 38, Average Number of Friends: 193.53
Age: 39, Average Number of Friends: 169.29
Age: 40, Average Number of Friends: 250.82
Age: 41, Average Number of Friends: 268.56
Age: 42, Average Number of Friends: 303.50
Age: 43, Average Number of Friends: 230.57
Age: 44, Average Number of Friends: 282.17
Age: 45, Average Number of Friends: 389.54
Age: 46, Average Number of Friends: 223.69
Age: 47, Average Number of Friends: 233.22
Age: 48, Average Number of Friends: 281.40
Age: 49, Average Number of Friends: 184.67
Age: 50, Average Number of Friends: 254.60
Age: 51, Average Number of Friends: 302.14
Age: 52, Average Number of Friends: 340.64
Age: 53, Average Number of Friends: 222.86
Age: 54, Average Number of Friends: 278.08
Age: 55, Average Number of Friends: 295.54
Age: 56, Average Number of Friends: 306.67
Age: 57, Average Number of Friends: 258.83
Age: 58, Average Number of Friends: 116.55
Age: 59, Average Number of Friends: 220.00
Age: 60, Average Number of Friends: 202.71
Age: 61, Average Number of Friends: 256.22
Age: 62, Average Number of Friends: 220.77
Age: 63, Average Number of Friends: 384.00
Age: 64, Average Number of Friends: 281.33
Age: 65, Average Number of Friends: 298.20
Age: 66, Average Number of Friends: 276.44
Age: 67, Average Number of Friends: 214.62
Age: 68, Average Number of Friends: 269.60
Age: 69, Average Number of Friends: 235.20

Age: 18, Max Friends: 499, Min Friends: 24
Age: 19, Max Friends: 404, Min Friends: 5
Age: 20, Max Friends: 384, Min Friends: 1
Age: 21, Max Friends: 491, Min Friends: 89
Age: 22, Max Friends: 478, Min Friends: 6
Age: 23, Max Friends: 392, Min Friends: 65
Age: 24, Max Friends: 492, Min Friends: 49
Age: 25, Max Friends: 485, Min Friends: 1
Age: 26, Max Friends: 492, Min Friends: 2
Age: 27, Max Friends: 471, Min Friends: 53
Age: 28, Max Friends: 378, Min Friends: 32
Age: 29, Max Friends: 367, Min Friends: 11
Age: 30, Max Friends: 487, Min Friends: 17
Age: 31, Max Friends: 481, Min Friends: 15
Age: 32, Max Friends: 412, Min Friends: 24
Age: 33, Max Friends: 471, Min Friends: 74
Age: 34, Max Friends: 423, Min Friends: 48
Age: 35, Max Friends: 428, Min Friends: 13
Age: 36, Max Friends: 493, Min Friends: 49
Age: 37, Max Friends: 471, Min Friends: 46
Age: 38, Max Friends: 459, Min Friends: 2
Age: 39, Max Friends: 275, Min Friends: 68
Age: 40, Max Friends: 465, Min Friends: 7
Age: 41, Max Friends: 397, Min Friends: 62
Age: 42, Max Friends: 467, Min Friends: 95
Age: 43, Max Friends: 428, Min Friends: 48
Age: 44, Max Friends: 499, Min Friends: 61
Age: 45, Max Friends: 497, Min Friends: 54
Age: 46, Max Friends: 462, Min Friends: 63
Age: 47, Max Friends: 488, Min Friends: 4
Age: 48, Max Friends: 439, Min Friends: 57
Age: 49, Max Friends: 476, Min Friends: 17
Age: 50, Max Friends: 436, Min Friends: 119
Age: 51, Max Friends: 493, Min Friends: 81
Age: 52, Max Friends: 487, Min Friends: 77
Age: 53, Max Friends: 451, Min Friends: 86
Age: 54, Max Friends: 462, Min Friends: 7
Age: 55, Max Friends: 474, Min Friends: 57
Age: 56, Max Friends: 444, Min Friends: 15
Age: 57, Max Friends: 465, Min Friends: 8
Age: 58, Max Friends: 348, Min Friends: 6
Age: 59, Max Friends: 439, Min Friends: 14
Age: 60, Max Friends: 324, Min Friends: 2
Age: 61, Max Friends: 469, Min Friends: 2
Age: 62, Max Friends: 496, Min Friends: 12
Age: 63, Max Friends: 469, Min Friends: 342
Age: 64, Max Friends: 499, Min Friends: 65
Age: 65, Max Friends: 443, Min Friends: 101
Age: 66, Max Friends: 496, Min Friends: 41
Age: 67, Max Friends: 445, Min Friends: 35
Age: 68, Max Friends: 490, Min Friends: 21
Age: 69, Max Friends: 491, Min Friends: 9

```
[ 'ITE00100554', '18000101', 'TMAX', '-75' ]
[ 'ITE00100554', '18000101', 'TMIN', '-148' ]
[ 'GM000010962', '18000101', 'PRCP', '0' ]
[ 'EZE00100082', '18000101', 'TMAX', '-86' ]
[ 'EZE00100082', '18000101', 'TMIN', '-135' ]
[ 'ITE00100554', '18000102', 'TMAX', '-60' ]
```

```
[ 'ITE00100554', '18000101', 'TMIN', '-148' ]
[ 'EZE00100082', '18000101', 'TMIN', '-135' ]
[ 'ITE00100554', '18000102', 'TMIN', '-125' ]
[ 'EZE00100082', '18000102', 'TMIN', '-130' ]
[ 'ITE00100554', '18000103', 'TMIN', '-46' ]
[ 'EZE00100082', '18000103', 'TMIN', '-73' ]
```

```

minimum temperature by itemID
('ITE00100554', -148)
Minimum temperature overall -148 ('EZE00100082', -135)

minimum temperature by StationID
('18000102', -130)
('18000104', -74)
('18000106', -57)
('18000110', -75)
('18000111', -62)
('18000112', -60)
('18000114', -35)
('18000115', -23)
('18000116', -37)
('18000117', -35)
('18000118', 9)

maximum temperature by StationID
('18000102', -44)
('18000104', 0)
('18000106', 13)
('18000110', 46)
('18000111', 66)
('18000112', 41)
('18000114', 41)
('18000115', 54)
('18000116', 56)
('18000117', 84)
('18000118', 59)

```

Result:

Successfully understood map value functions and utilised them for various tasks

Ex. No: 7	
27th August, 2024	

Hadoop and Docker

Aim:

To install hadoop in docker containers

Output:

The terminal window shows the following content:

```
└── ex_docker
    ├── Dockerfile
    ├── dummy
    ├── hadoop-streaming-3.4.0.jar
    ├── mapper.py
    ├── reducer.py
    ├── wc.jar
    ├── WordCount.class
    ├── WordCount.java
    ├── WordCount$IntSumReducer.class
    ├── WordCount$TokenizerMapper.class
    └── words.txt

SUBCOMMAND may print help when invoked w/o parameters or with -h.
hadoop@85dfacaf27ee:~/ex_docker$ hdfs dfs -cat /user/hadoop/output/part-r-00000
I          1
a          1
above     1
are       1
diamond   1
high      1
how       1
in        1
like      1
little    2
sky       1
so        1
star      2
the       2
twinkle   4
up        1
what      1
wonder   1
workd    1
you      1
hadoop@85dfacaf27ee:~/ex_docker$ s
```

Name	Image	Status	Port(s)	CPU (%)	Last started	Actions
hadoop-container 85dfacaf27ee	hadoop-image:0.0.1	Running		3.16%	1 hour ago	Stop Logs Open

Result:

Successfully installed hadoop in docker containers

Ex. No: 8	Public and Private Keys
3rd September, 2024	

Aim:

To secure an EC2 instance using SSH Keys and Network Access Control.

Program:

1. Generate Private and Public Key Pairs

- o Linux/Mac:

- Create a new directory for key pairs: `mkdir key-pair-labs && cd key-pair-labs`
- Generate a private key: `openssl genrsa -out snu-privatekey.pem 2048`
- Generate a public key from the private key: `openssl rsa -in snu-privatekey.pem -pubout -out snu-publickey.pem`
- Set permissions on the private key: `chmod 400 snu-privatekey.pem`
- Copy the public key: `cat snu-publickey.pem`

- o Paste the public key into the AWS console.

2. Launch an Ubuntu EC2 Instance:

- o Launch a new EC2 instance using the public key.
- o Choose an appropriate instance type and AMI.
- o Allocate an Elastic IP address and attach it to the instance.

3. Login to the Instance:

- o Use the private key generated in step 1 to SSH into the instance.
- o Linux / Mac : `ssh -i snu-privatekey.pem ubuntu@<public_ip_address>`

4. Edit Security Group:

- o Open the Security Group settings for the instance.
- o Add an inbound rule to allow ICMP traffic (ping) from anywhere:
 - Type: Custom TCP Rule
 - Protocol: ICMP
 - Port Range: All
 - Source: 0.0.0.0/0

5. Ping the Instance:

- Ping the public IP address of the instance to verify connectivity.

- You should be able to ping the instance successfully.

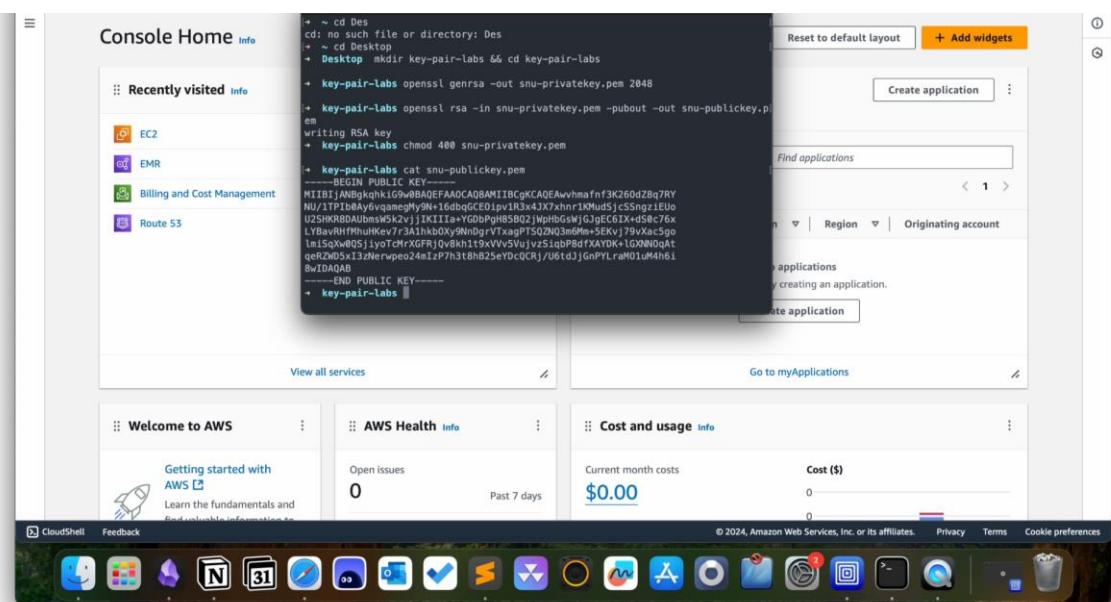
6. Edit Default NACL:

- Open the Network Access Control List (NACL) settings for the instance.
- Edit the default NACL to block ICMP traffic.
- Add an egress rule to deny ICMP traffic:
 - Type: Custom TCP Rule
 - Protocol: ICMP
 - Port Range: All
 - Destination: 0.0.0.0/0

7. Ping the Instance (Again):

- Ping the public IP address of the instance again.
- You should not be able to ping the instance this time, as ICMP traffic is now blocked.

Output:



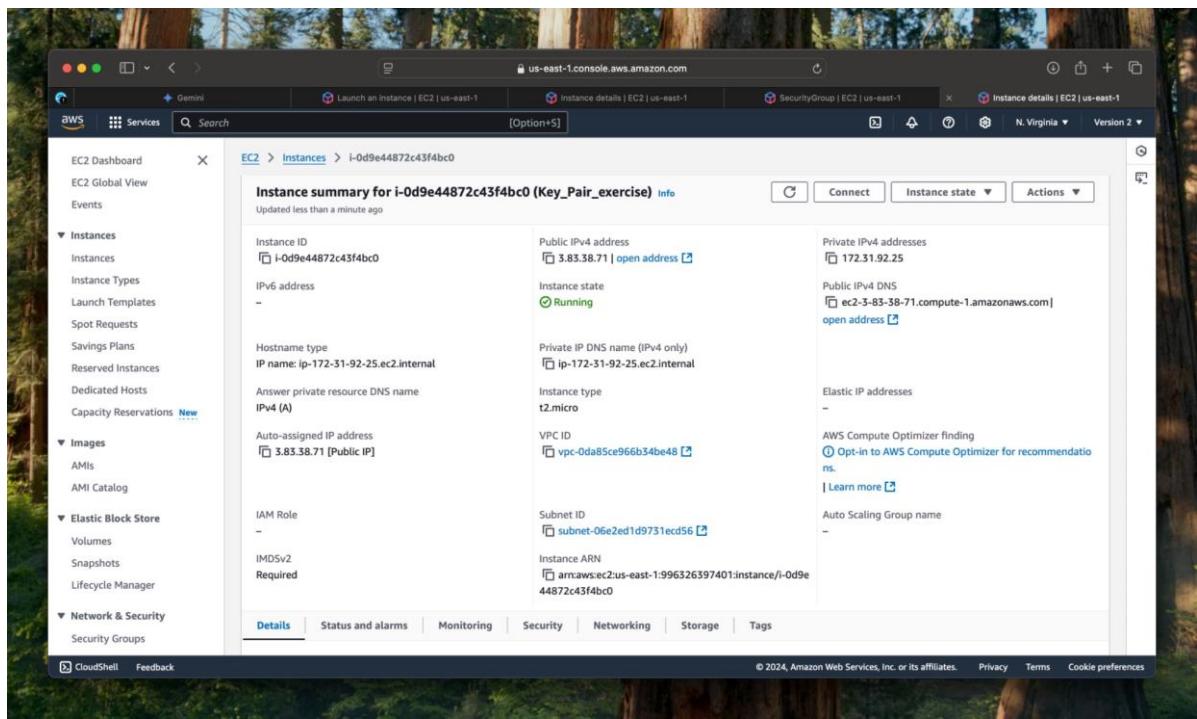
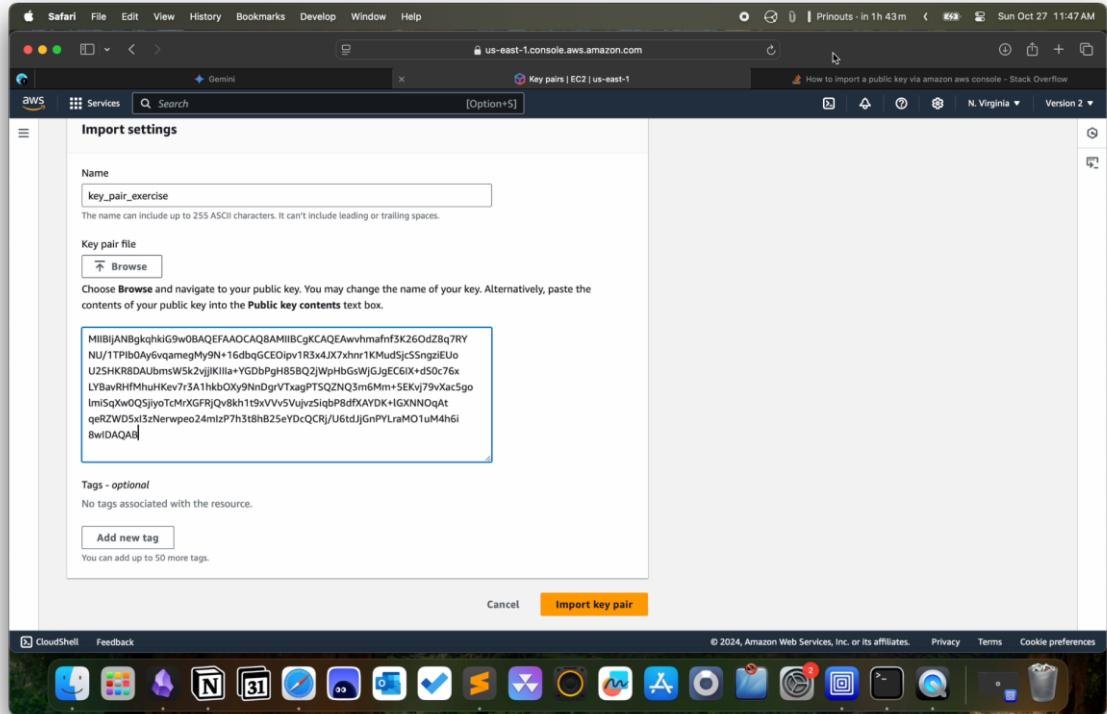
The screenshot shows the AWS CloudShell interface with a terminal window open. The terminal displays the following command sequence and its output:

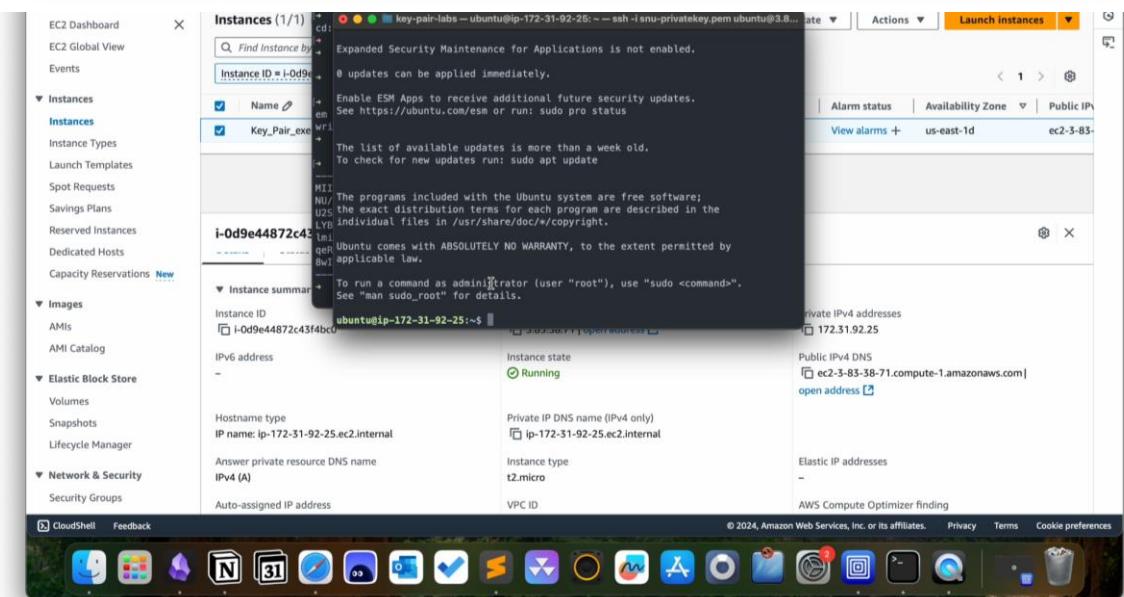
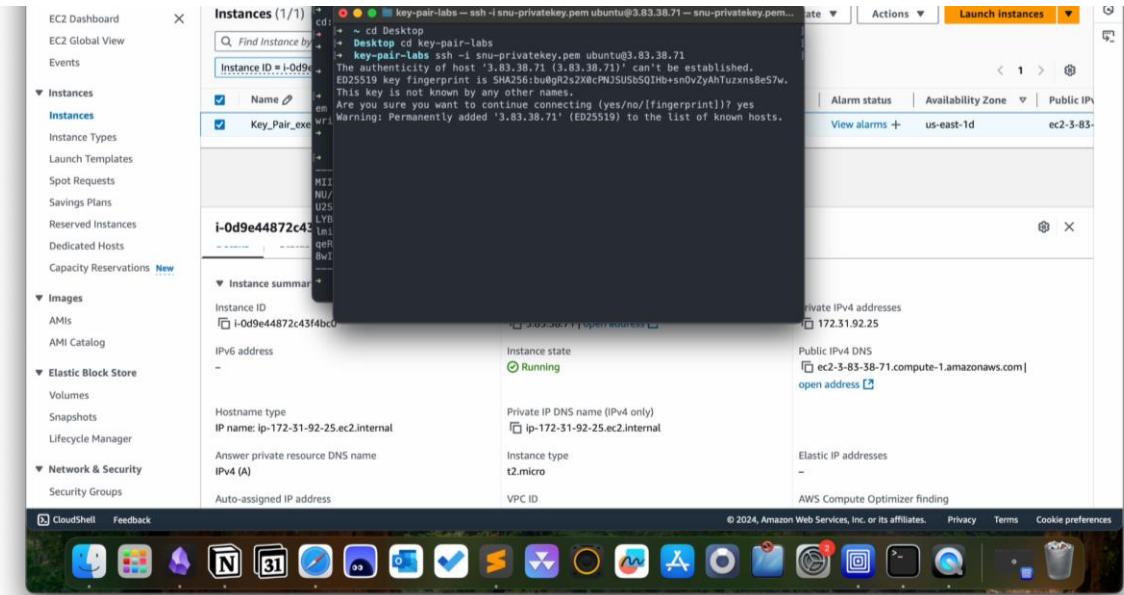
```

cd: no such file or directory: Des
~ cd Desktop
+ Desktop mkdir key-pair-labs && cd key-pair-labs
+ key-pair-labs openssl genrsa -out snu-privatekey.pem 2048
+ key-pair-labs openssl rsa -in snu-privatekey.pem -pubout -out snu-publickey.pem
writing RSA key
+ key-pair-labs chmod 400 snu-privatekey.pem
+ key-pair-labs cat snu-publickey.pem
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AII8CgKCAQEAvvhmaFnf3K260dZ8q7RY
NU/1TP1bAay6vqamegy5N+16gbGCE01pv1r3x4JX7xhnr1K0udSj+c5ngz1EUo
U2SHKRBD0AbJbnsW5k2vjj1K1Ii+YGD0PgH8SB02jWpBGSwJGjgEC6Ix+d5c=76x
LYBaVRffmhuKev7/3A1hkOx9N0DgrTxagfTS02Q3mfMm+5EKvJ79vKac5go
lm15qXw@0SjIyoTMrXGFjRjQvkh1t9xVv5Vjyvz1qbP8dfXYADKjLGXN0qAt
qeRZh05x13Nerwpeo24mIzP7h3t8hB25eyDc0CRj/U6tdJJGnPylraM01uM4h61
8wIDAQAB
-----END PUBLIC KEY-----
+ key-pair-labs

```

The terminal window has a dark background with white text. The AWS navigation bar is visible at the top, and the CloudShell toolbar is at the bottom.





Safari File Edit View History Bookmarks Develop Window Help

us-east-1.console.aws.amazon.com

Gemini Launch an instance | EC2 | us-east-1 Instances | EC2 | us-east-1 ModifyInboundSecurityGroupRules | EC2 | us-east-1 N. Virginia Version 2

Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Security group rule ID	Type <small>Info</small>	Protocol <small>Info</small>	Port range	Source <small>Info</small>	Description - optional <small>Info</small>
sgr-01f3f2dcce314d76e	SSH	TCP	22	Custom	<input type="text"/> 0.0.0.0/0 <small>X</small>
-	All ICMP - IPv4	ICMP	All	Anyw...	<input type="text"/> 0.0.0.0/0 <small>X</small>
-	All ICMP - IPv6	IPv6 ICMP	All	Anyw...	<input type="text"/> 0.0.0.0/0 <small>X</small>
					<input type="text"/> 0.0.0.0/0 <small>X</small>

Add rule

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. X

Cancel Preview changes Save rules

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Terminal Shell Edit View Window Help

us-east-1.console.aws.amazon.com

Gemini Launch an instance | EC2 | us-east-1 Instance details | EC2 | us-east-1 SecurityGroup | EC2 | us-east-1 N. Virginia Version 2

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EC2 Dashboard EC2 Global View Events

Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Capacity Reservations New

Images AMIs AMI Catalog

Elastic Block Store Volumes Snapshots Lifecycle Manager

Network & Security Security Groups

Instance summary Updated less than a minute ago

Instance ID i-0d9e44872

IPV6 address -

Hostname type IP name: ip-172-31-10-108

IPV4 (A) Auto-assigned IP 3.83.38.71

```
+ key-pair-labs openssl genrsa -out snu-privatekey.pem 2048
+ key-pair-labs openssl rsa -in snu-privatekey.pem -pubout -out snu-publickey.pem
writing RSA key
+ key-pair-labs chmod 400 snu-privatekey.pem
+ key-pair-labs cat snu-publickey.pem
-BEGIN PUBLIC KEY-
MIIBIjANBgkqhkiG9w0BAQEFAAQABMIIIBGKCAQEAvvhmafNf3K260d28q7RY
NU/1Bb8y6vamgMyWn+16fbzGCE0ipv1R3x4JX7hn1K0udSj+S5nqZlE0
U2SHKR8DAlbnwsNSk2yjj1K1II+xDGdpH8SB02jWpHbsGw[G]qEc6IX+d5c76x
LYBa8r0fMhuHKev7r3A1hk0Xy9NbnpgrTxagPTs02N0mMm+5EkV79xAc5g
lm5Qxw@005jlyoTMrXGRj0v8k1t9xVv5Vjvz5qbP8dfXAYDK-1gXNNoGat
qeRZ0W5x13zNevrpeo24mIz7p7h3t8hB25eY0CCRj/U6tdjGnPfLrM01uM4h61
+ key-pair-labs ping 3.83.38.71
PING 3.83.38.71 (3.83.38.71): 56 data bytes
64 bytes from 3.83.38.71: icmp_seq=0 ttl=47 time=831.556 ms
Request timeout for icmp_seq 1
64 bytes from 3.83.38.71: icmp_seq=1 ttl=47 time=1010.987 ms
64 bytes from 3.83.38.71: icmp_seq=2 ttl=47 time=226.695 ms
```

Connect Instance state Actions

Private IPv4 addresses 172.31.92.25

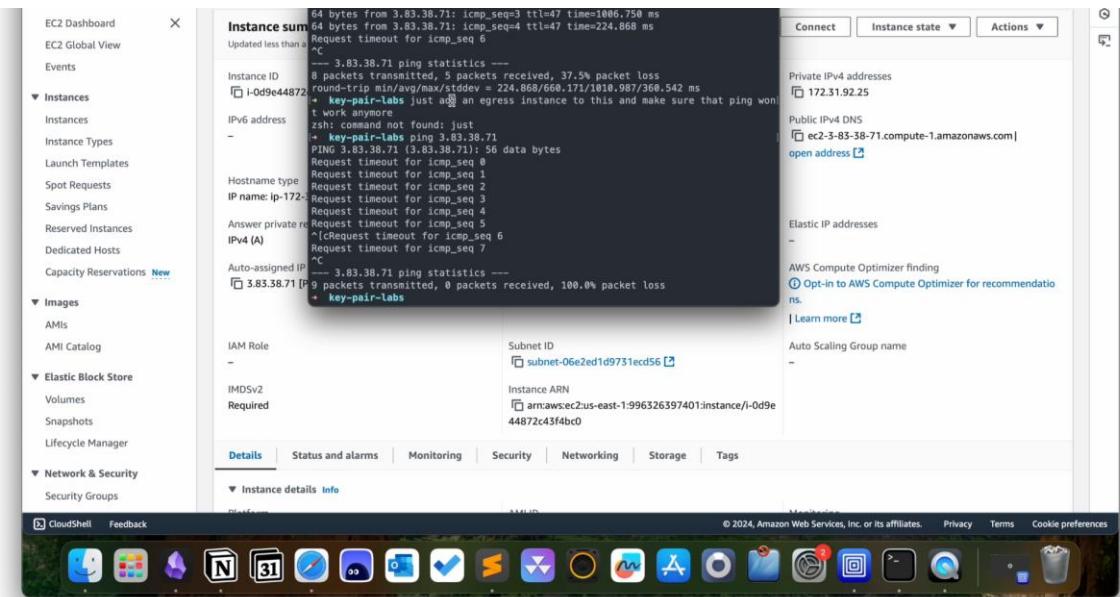
Public IPv4 DNS ec2-3-83-38-71.compute-1.amazonaws.com | open address Link

Elastic IP addresses -

AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations Learn more

Auto Scaling Group name -

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



Result:

Successfully Created Public and Private Keys and Utilised it to secure EC2 instances.

Ex. No: 9	
10th September, 2024	EC2

Aim:

To create an EC2 instance and Deploy a web server on it using Apache2

Program:

- 1. Create a Key Pair:**
 - o Use the AWS Management Console to create a new key pair.
 - o Download the private key file (.pem) and save it securely.
- 2. Launch EC2 Instance:**
 - o Ubuntu AMI,
 - o Choose an instance type based on your requirements.
 - o Configure security groups to allow SSH and HTTP traffic.
 - o Launch the instance and provide the private key file during the launch process.
- 3. Connect to Instance:**
 - o Use the SSH client to connect to the instance using the public IP address and private key file.
- 4. Update Package Lists:**
 - o Run the following command to update the package lists : sudo apt update
- 5. Install Apache2:** sudo apt install apache2
- 6. Check Apache Status:** sudo service apache2 status
- 7. Test Apache:**
 - o Open a web browser and enter the public IP address of the instance.
 - o You should see the default Apache2 welcome page.
- 8. Change the Contents of the Webpage:**
 - o cd /var/www/html/
 - o sudo chmod 777 index.html
 - o echo "website" > index.html
- 9. Test the website to show new content**

Output:

Instance summary for i-0ba1ed29c2be3ef84 (EC2_Lab)

Instance ID	i-0ba1ed29c2be3ef84	Public IPv4 address	184.73.151.28 [open address]	Private IPv4 addresses	172.31.93.220
IPv6 address	-	Instance state	Running	Public IPv4 DNS	ec2-184-73-151-28.compute-1.amazonaws.com [open address]
Hostname type	IP name: ip-172-31-93-220.ec2.internal	Private IP DNS name (IPv4 only)	ip-172-31-93-220.ec2.internal	Elastic IP addresses	-
IP name	172.31.93.220.ec2.internal	Instance type	t2.micro	AWS Compute Optimizer finding	Opt-in to AWS Compute Optimizer for recommendations Learn more
Answer private resource DNS name	IPv4 (A)	VPC ID	vpc-0da85ce966b34be48	Auto Scaling Group name	-
Auto-assigned IP address	184.73.151.28 [Public IP]	Subnet ID	subnet-06e2ed1d9731ecd56		
IAM Role	-	Instance ARN	arn:aws:ec2:us-east-1:996326397401:instance/i-0ba1ed29c2be3ef84		
IMDSv2	Required				

```

key-pair-labs — ubuntu@ip-172-31-93-220: ~ -- ssh -i snu-privatekey.pem ubuntu@1...
→ Desktop mkdir key-pair-labs && cd key-pair-labs

→ key-pair-labs openssl genrsa -out snu-privatekey.pem 2048

[→ key-pair-labs openssl rsa -in snu-privatekey.pem -pubout -out snu-publickey.pem
em
writing RSA key
→ key-pair-labs chmod 400 snu-privatekey.pem

[→ key-pair-labs cat snu-publickey.pem
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAQ8AMIIBCgKCAQEAn5V9u25EVk5nNfs0qMoi
shWlxxwjP9A1MGYOHIQjQdInUdk1t4U1NR57m+9YlRwaYyYojiMen1IJcVdHGzTl
R3BbGmPSKhR7ilawYKwb3IoCP3s1NPbg/Mq88PjzEdKPPjUP9xisnnNCkLNkoTwo
tnpXEtWUXdJ/8pWvwGaVQsUWHjN8YHbL3/oRoVndhWps9BGn4cqRV1DMpq5Qwkki
+iApDy4sj3xg24fha0pnbgZDYJSikarqLvT4TVISuNqSuxwj0qoMGZhkmold8Tcy
I873xrMgcAmmxwRfQ3FAhD39dRrootZPNvhDL0TmjEFVee1RpVrs5QDyFCRVa08Y
uwIDAQAB
-----END PUBLIC KEY-----
[→ key-pair-labs ssh -i snu-privatekey.pem ubuntu@184.73.151.28
The authenticity of host '184.73.151.28 (184.73.151.28)' can't be established.
ED25519 key fingerprint is SHA256:0447y6SkAs3URJwfWj4XN8W9P1GDvUeMPMA04QdgfnY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
]

```

```
key-pair-labs — ubuntu@ip-172-31-93-220: ~ — ssh -i snu-privatekey.pem ubuntu@1...
[ubuntu@ip-172-31-93-220:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [12
6 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packag
es [15.0 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-
en [5982 kB]
Get:7 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [433
kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Compon
ents [3871 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f
Metadata [301 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Pac
kages [269 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translati
on-en [118 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Com
ponents [35.0 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-
```

```
key-pair-labs — ubuntu@ip-172-31-93-220: ~ — ssh -i snu-privatekey.pem ubuntu@1...
[ubuntu@ip-172-31-93-220:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libaprutil1t64 liblua5.4-0 ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1t64
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 liblua5.4-0 ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 28 not upgraded.
Need to get 2084 kB of archives.
After this operation, 8094 kB of additional disk space will be used.
[Do you want to continue? [Y/n] Y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 li
bapr1t64 amd64 1.7.2-3.1ubuntu0.1 [108 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil
1t64 amd64 1.6.3-1.1ubuntu7 [91.9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil
1-dbd-sqlite3 amd64 1.6.3-1.1ubuntu7 [11.2 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil
1-ldap amd64 1.6.3-1.1ubuntu7 [9116 B]
```

```

key-pair-labs — ubuntu@ip-172-31-93-220: ~ — ssh -i snu-privatekey.pem ubuntu@1...
No services need to be restarted.

No containers need to be restarted.

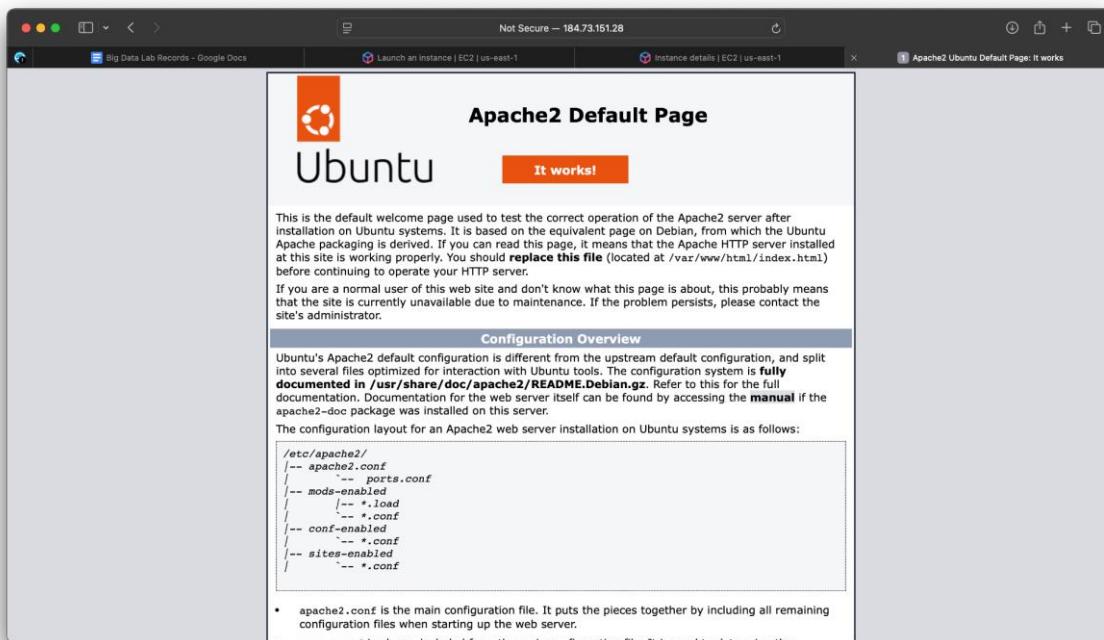
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.

[ubuntu@ip-172-31-93-220:~$ sudo service apache2 status
● apache2.service - The Apache HTTP Server
    Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: ▶)
    Active: active (running) since Sun 2024-10-27 07:58:20 UTC; 14s ago
      Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2060 (apache2)
     Tasks: 55 (limit: 1130)
    Memory: 5.4M (peak: 5.7M)
       CPU: 32ms
      CGroup: /system.slice/apache2.service
              ├─2060 /usr/sbin/apache2 -k start
              ├─2063 /usr/sbin/apache2 -k start
              └─2064 /usr/sbin/apache2 -k start

Oct 27 07:58:20 ip-172-31-93-220 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Oct 27 07:58:20 ip-172-31-93-220 systemd[1]: Started apache2.service - The Apache HTTP Server...
lines 1-15/15 (END)

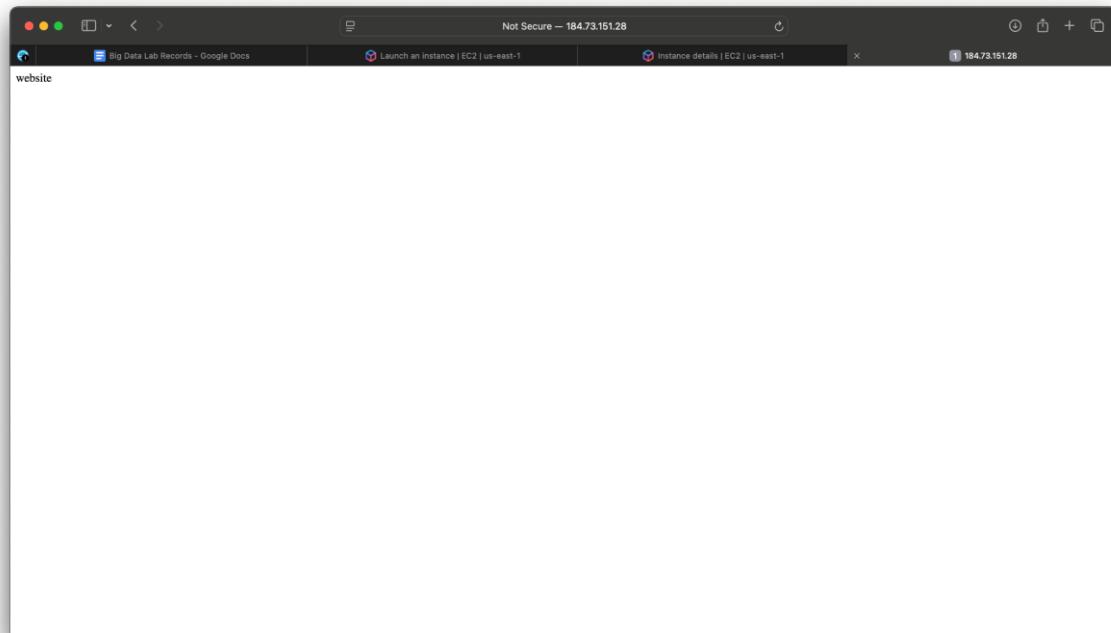
```



```
key-pair-labs — ubuntu@ip-172-31-93-220: /var/www/html — ssh -i snu-privatekey.p...
21 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Sun Oct 27 07:57:17 2024 from 49.43.251.85
[ubuntu@ip-172-31-93-220:~$ cd /var/www/html/
[ubuntu@ip-172-31-93-220:/var/www/html$ suo chmod 777 index.html
Command 'suo' not found, did you mean:
  command 'su' from deb util-linux (2.39.3-9ubuntu6.1)
  command 'sur' from deb subtle (0.11.3224-xi-2.2build5)
  command 'sup' from deb sup (20100519-3)
  command 'sul' from deb hxtools (20231101-1)
  command 'sudo' from deb sudo (1.9.14p2-1ubuntu1)
  command 'sudo' from deb sudo-ldap (1.9.14p2-1ubuntu1)
  command 'sumo' from deb sumo (1.18.0+dfsg-3build2)
  command 'zuo' from deb zuo (1.9-1)
  command 'sum' from deb coreutils (9.4-2ubuntu2)
Try: sudo apt install <deb name>
[ubuntu@ip-172-31-93-220:/var/www/html$ sudo chmod 777 index.html
[ubuntu@ip-172-31-93-220:/var/www/html$ echo "website" > index.html
ubuntu@ip-172-31-93-220:/var/www/html$
```



Result:

Successfully deployed an EC2 instance and tested using Apache Webservices.

Ex. No: 10	Route53
1st October, 2024	

Aim:

To set up a web server on an AWS EC2 instance and configure a domain name using GoDaddy and Route 53

Program:

1. **Login to GoDaddy and Create a subdomain:**
 - o Use your 12-digit registration number.
 - o Example: 21100101001.ngaws.xyz
2. **Create a Hosted Zone in Route 53:**
 - o Navigate to the Route 53 service in the AWS console.
 - o Create a new Hosted Zone for your subdomain.
3. **Get the Name Server information:**
 - o Retrieve the name server information from the Route 53 dashboard.
 - o Example: ns-1234.awsdns-12.org, [invalid URL removed]
4. **Update the NS record in GoDaddy:**
 - o Log in to the GoDaddy portal.
 - o Navigate to the DNS settings for your subdomain.
 - o Update the NS records with the name servers obtained from Route 53.
5. **Create an EC2 instance:**
 - o Launch an EC2 instance with an elastic public IP address.
6. **Install Apache:**
 - o Connect to the EC2 instance using SSH.
 - o Update the package lists: sudo apt update
 - o Install Apache: sudo apt install apache2
7. **Configure Apache:**
 - o Edit the Apache configuration file: sudo nano /etc/apache2/sites-available/000-default.conf
 - o Modify the DocumentRoot and ServerName directives to point to your desired web content directory and domain name.
 - o Save the configuration file and restart Apache: sudo systemctl restart apache2

8. Create an A record in Route 53:

- o Navigate to your Hosted Zone in Route 53.
- o Create a new A record with the following details:
 - Name: @ (for the root domain)
 - Value: The public IP address of your EC2 instance
 - TTL: 3600 (1 hour)

Output:

The screenshot shows the AWS Route 53 service interface. On the left, a sidebar navigation includes 'Route 53', 'Dashboard', 'Hosted zones' (selected), 'Health checks', 'Profiles New', 'IP-based routing', 'Traffic flow', 'Domains', and 'Resolver'. The main content area displays a success message: '21011101055.ngaws.xyz was successfully created. Now you can create records in the hosted zone to specify how you want Route 53 to route traffic for your domain.' Below this, the 'Hosted zones' section shows '21011101055.ngaws.xyz' (Public). The 'Hosted zone details' tab is selected, showing 'Records (2)'. The table lists two records:

Type	Name	Value	TTL	Alias	Value/Route traffic to
NS	21011101...	ns-124.awsdns-15.com.	600 seconds	No	ns-999.awsdns-60.net. ns-2045.awsdns-63.co.uk. ns-1475.awsdns-56.org.
SOA	21011101...	ns-124.awsdns-15.com. awsd...	600 seconds	No	ns-124.awsdns-15.com.

The screenshot shows the GoDaddy DNS Management interface. The left sidebar includes 'Domains', 'Portfolio', 'DNS' (selected), 'Transfers', 'Services', 'Tools', and 'Settings'. The main content area shows a table of DNS records. A success message box is displayed in the top right corner: 'Your DNS record has been updated successfully. Most DNS updates take effect within an hour, but could take up to 48 hours to update globally.' The table lists the following records:

Type	Name	Value	TTL	Actions	
A	jenkins	13.233.133.184	600 seconds		
A	varun-todo	3.108.220.63	600 seconds		
NS	@	ns51.domaincontrol.com.	1 Hour	Can't delete	Can't edit
NS	@	ns52.domaincontrol.com.	1 Hour	Can't delete	Can't edit
NS	21011101055	ns-124.awsdns-15.com.	1 Hour		
NS	21011101074	ns-116.awsdns-14.com.	1 Hour		
NS	21011101075	ns-127.awsdns-15.com.	1 Hour		
NS	21011101079	ns-289.awsdns-36.com.	1 Hour		
NS	21011101079	ns-510.awsdns-63.com.	1 Hour		

The screenshot shows the AWS EC2 Instances details page for instance `i-09a90d79acc628dbd`. The instance is currently running. Key details include:

- Public IPv4 address:** 54.162.240.174
- Private IP4 addresses:** 172.31.45.111
- Public IPv4 DNS:** ec2-54-162-240-174.compute-1.amazonaws.com
- Instance type:** t2.micro
- VPC ID:** vpc-0da85ce966b34be48
- Subnet ID:** subnet-08c820c8826320fef
- Instance ARN:** arn:aws:ec2:us-east-1:996326397401:instance/i-09a90d79acc628dbd

The page also includes tabs for Status and alarms, Monitoring, Security, Networking, Storage, and Tags.

The screenshot shows the AWS Route 53 Hosted Zone details page for the zone `21011101055.ngaws.xyz`. A success message indicates the record was created. The hosted zone details section shows:

- Records (3):** ns-124.awsdns-15.com., ns-999.awsdns-60.net., ns-2045.awsdns-63.co.uk., ns-1475.awsdns-56.org.
- SOA Record:** 21011101..., SOA, Simple, -
- A Record:** www.210..., A, Simple, -

The page includes tabs for Delete zone, Test record, and Configure query logging.

```
[→ ~ links www.21011101055.ngaws.xyz  
zsh: command not found: links  
[→ ~ nslookup www.21011101055.ngaws.xyz  
Server:      172.16.0.2  
Address:      172.16.0.2#53  
  
Non-authoritative answer:  
Name:    www.21011101055.ngaws.xyz  
Address: 54.162.240.174  
  
→ ~ ]
```

Result:

Successfully implemented a Route53 DNS Lookup on AWS.

Ex. No: 11	IAM
8th October, 2024	

Aim:

To implement IAM user access in AWS Console.

Program:

1. Create User Groups server_admin and dns_admin. Give them full access to EC2 Service and Route53 services.
2. Create users, alice, bob, cathay and david. Set passwords for them
3. Add alice and bob to user group server_adin and cathy and david to user group dns_admin.
4. Create user eve and give him Billing access.
5. Create user hadoop and give full (admin) access to the services.
6. Create an alias name for your account.
7. Use the alias URL to login to your account instead of the account ID.

Output:

Screenshot of the AWS IAM User Groups creation page.

User group name: dns_admin

Add users to the group - Optional (0)

Attach permissions policies - Optional (1/959)

Name	Type	Used as	Description
AmazonRoute53FullA...	AWS managed	None	Provides full access to all Amazon Rou...

Create user group

Screenshot of the AWS IAM User creation review step.

Review and create

User details

User name	Console password type	Require password reset
alice	Autogenerated	Yes

Permissions summary

Name	Type	Used as
IAMUserChangePassword	AWS managed	Permissions policy
server_admin	Group	Permissions group

Tags - optional

No tags associated with the resource.

Add new tag

Screenshot of the AWS Management Console showing the IAM Users page after creating a new user.

The browser address bar shows: us-east-1.console.aws.amazon.com

The IAM sidebar menu is open, showing:

- Identity and Access Management (IAM)
- Dashboard
- Access management
 - User groups
 - Users**
 - Roles
 - Policies
 - Identity providers
 - Account settings
- Access reports
 - Access Analyzer
 - External access
 - Unused access
 - Analyzer settings
- Credential report
- Organization activity
- Service control policies

The main content area displays a success message: "User created successfully". It includes a link to "View user" and a "Create user" button. The "Users (6) Info" table lists six users:

User name	Path	Group	Last activity	MFA	Password age	Console last sign-in
alice	/	1	-	-	-	-
bob	/	1	-	-	-	-
cathy	/	1	-	-	-	-
david	/	1	-	-	-	-
eve	/	0	-	-	-	-
vishwa	/	0	-	-	-	-

Page footer: © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the AWS sign-in page for IAM user sign-in.

The browser address bar shows: eu-north-1.signin.aws.amazon.com

A message box at the top left says: "You are currently using the improved sign in UI experience. The improved sign in experience will launch soon. During this time, you can still change back to legacy sign in using the dropdown in the upper right corner." It includes a close button (X).

The AWS logo is centered above the sign-in form.

The sign-in form fields are as follows:

- Account ID (12 digits) or account alias: metabaap
- IAM username: alice
- Password: 0znV8Th\$
- Show Password checkbox (checked)
- Having trouble? link
- Sign In button
- Sign in using root user email link
- Create a new AWS account link
- Remember this account checkbox

To the right of the sign-in form is a promotional banner for Amazon Lightsail, featuring a cartoon robot character giving a thumbs up.

You must change your password to continue

AWS account 996326397401

IAM user name alice

Old password _____

New password _____

Retype new password _____

Confirm password change

[Sign in using root user email](#)

English

[Terms of Use](#) [Privacy Policy](#) © 1996-2024, Amazon Web Services, Inc. or its affiliates.

eu-north-1.console.aws.amazon.com

Services [Option+S]

Reset to default layout

Applications (0) Create application

Region: Europe (Stockholm)

eu-north-1 (Current Region)

Name	Description	Region	Originating account
eu-north-1	Access denied	Europe (Stockholm)	alice @ metabaap

No recently visited services

Explore one of these commonly visited AWS services.

[EC2](#) [S3](#) [RDS](#) [Lambda](#)

[View all services](#) [Go to myApplications](#)

Welcome to AWS

Getting started with AWS

Learn the fundamentals and find valuable information to get the most out of AWS.

AWS Health

Cost and usage

Current month costs

Forecasted month end costs

Cost breakdown

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Result:

Successfully implemented IAM labs in AWS