HPE CTY WEEKLY PROGRESS DOCUMENTATION

CAPACITY ADVISOR FOR MONITORING RESOURCES

Week 1: Setting Up Apache Kafka and Prometheus on AWS EC2 (Linux)

- Installed and configured **Apache Kafka**, a distributed event streaming platform designed for high-throughput, fault-tolerant real-time data processing.
- Installed and configured **Prometheus**, an open-source monitoring and alerting toolkit optimized for time-series data and system performance metrics.

  Steps followed:

  APACHE KAFKA

  Pre-requisites

  - ➢ Launch an EC2 instance and allow port 22 for SSH.
  - ➢ Launch Putty and connect to the AWS EC2 instance:
    - ○ Install PuTTY (if not installed)
    - ○ Create the EC2 instance with a key pair (.pem file).
    - ○ Convert `.pem` to `.ppk` (Using PuTTYgen)
    - ○ Connect to AWS EC2 using PuTTY:
      - ■ Open PuTTY.
      - ■ In Host Name, enter – ec2-user@your-aws-public-ip
      - ■ In Connection > SSH > Auth, load the `.ppk` file under "Private key
      - ■ file for authentication".
      - ■ Click Open.
      - ■ Click "Yes" if you get a security alert.
  - ➢ Update Your System:
    - ○ sudo yum update -y
  - ➢ Install Java (Required for Kafka):
    - ○ sudo amazon-linux-extras enable corretto8
    - ○ sudo yum install java-11-amazon-corretto -y

  Downloading and Installing Kafka

  - ➢ Navigate to the `/opt` directory:
    - ○ cd /opt
  - ➢ Download Kafka:
    - ○ wget https://downloads.apache.org/kafka/3.6.1/kafka_2.13-3.6.1.tgz
  - ➢ Extract Kafka:
    - ○ sudo tar -xvzf kafka_2.13-3.6.1.tgz
    - ○ sudo mv kafka_2.13-3.6.1 kafka
  - ➢ Change ownership and permissions:
    - ○ sudo chown -R ec2-user:ec2-user /opt/kafka

  Start Zookeeper (Required for Kafka)

  - ➢ Start Zookeeper:
    - ○ cd /opt/kafka
    - ○ bin/zookeeper-server-start.sh config/zookeeper.properties

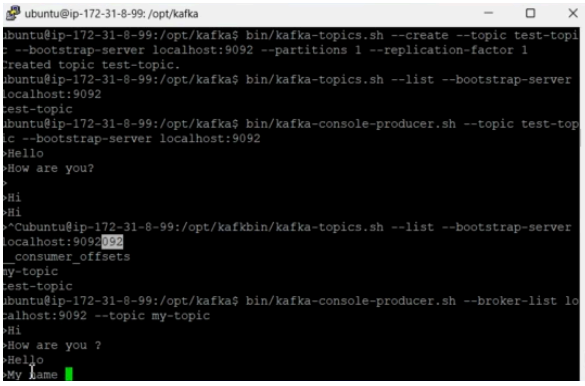
- ➢ Run Zookeeper in the background:
    - ○ nohup bin/zookeeper-server-start.sh config/zookeeper.properties > zookeeper.log

Start Kafka Broker

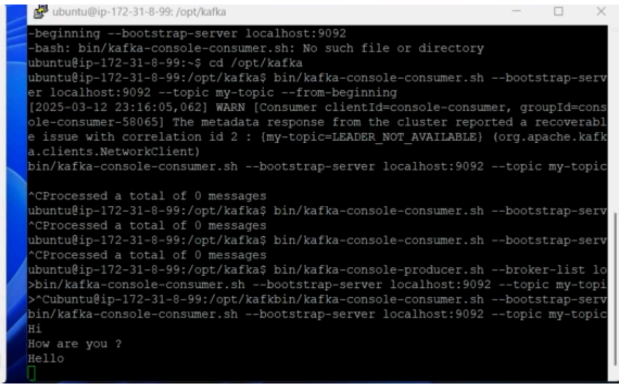
- ➢ Start Kafka:
    - ○ bin/kafka-server-start.sh config/server.properties
- ➢ Run Kafka in the background:
    - ○ nohup bin/kafka-server-start.sh config/server.properties > kafka.log 2>&1 &

Testing Kafka

- ➢ Create a Topic:
    - ○ bin/kafka-topics.sh --create --topic test-topic --bootstrap-server localhost:9092
    - ○ --partitions 1 --replication-factor 1
- ➢ Start a Kafka Producer to send messages
    - ○ bin/kafka-console-producer.sh --topic test-topic --bootstrap-server localhost:9092
- ➢ Start a Kafka Consumer to read messages
    - ○ bin/kafka-console-consumer.sh --topic test-topic --from-beginning --bootstrap-server
    - ○ localhost:9092



```
ubuntu@ip-172-31-8-99:/opt/kafka$ bin/kafka-topics.sh --create --topic test-topi
c --bootstrap-server localhost:9092 --partitions 1 --replication-factor 1
Created topic test-topic.
ubuntu@ip-172-31-8-99:/opt/kafka$ bin/kafka-topics.sh --list --bootstrap-server
localhost:9092
test-topic
ubuntu@ip-172-31-8-99:/opt/kafka$ bin/kafka-console-producer.sh --topic test-top
ic --bootstrap-server localhost:9092
>Hello
>How are you?
>
>Hi
>Hi
>^Cubuntu@ip-172-31-8-99:/opt/kafkbin/kafka-topics.sh --list --bootstrap-server
localhost:9092092
__consumer_offsets
my-topic
test-topic
ubuntu@ip-172-31-8-99:/opt/kafka$ bin/kafka-console-producer.sh --broker-list lo
calhost:9092 --topic my-topic
>Hi
>How are you ?
>Hello
>My name
```

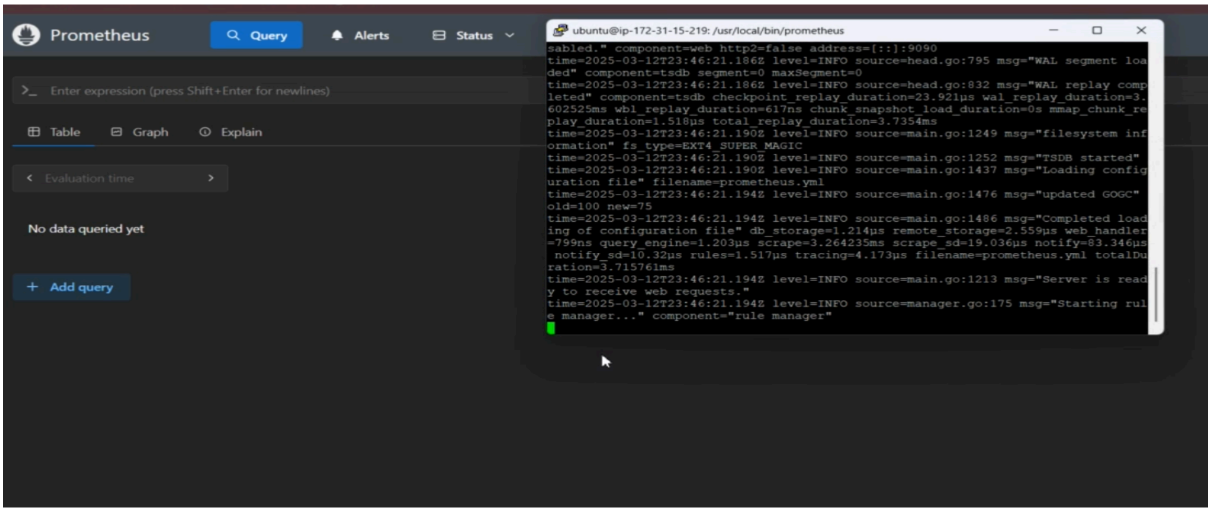


```
-beginning --bootstrap-server localhost:9092
-bash: bin/kafka-console-consumer.sh: No such file or directory
ubuntu@ip-172-31-8-99:~$ cd /opt/kafka
ubuntu@ip-172-31-8-99:/opt/kafka$ bin/kafka-console-consumer.sh --bootstrap-serv
er localhost:9092 --topic my-topic --from-beginning
[2025-03-12 23:16:05,062] WARN [Consumer clientId=console-consumer, groupId=cons
ole-consumer-58065] The metadata response from the cluster reported a recoverabl
e issue with correlation id 2 : {my-topic=LEADER_NOT_AVAILABLE} (org.apache.kafk
a.clients.NetworkClient)
bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic my-topic

^CProcessed a total of 0 messages
ubuntu@ip-172-31-8-99:/opt/kafka$ bin/kafka-console-consumer.sh --bootstrap-serv
^CProcessed a total of 0 messages
ubuntu@ip-172-31-8-99:/opt/kafka$ bin/kafka-console-consumer.sh --bootstrap-serv
^CProcessed a total of 0 messages
ubuntu@ip-172-31-8-99:/opt/kafka$ bin/kafka-console-producer.sh --broker-list lo
>bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic my-topi
>^Cubuntu@ip-172-31-8-99:/opt/kafkbin/kafka-console-consumer.sh --bootstrap-serv
bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic my-topic
Hi
How are you ?
Hello
```

PROMETHEUS

- ➢ Launch an EC2 Instance and allow port 22 for SSH, port 9090 for Prometheus, and port 9100 for Node Exporter.
- ➢ Update system packages:
    - ○ sudo apt update && sudo apt upgrade -y
- ➢ Download Prometheus: wget https://github.com/prometheus/prometheus/releases/download/v3.2.1/prometheus-3.2.1.linux-amd64.tar.gz
- ➢ Extract and move files:
    - ○ tar -xvzf prometheus-3.2.1.linux-amd64.tar.gz
    - ○ sudo mv prometheus-3.2.1.linux-amd64 /etc/prometheus
- ➢ Create Prometheus user and set permissions:
    - ○ sudo useradd --no-create-home --shell /bin/false prometheus

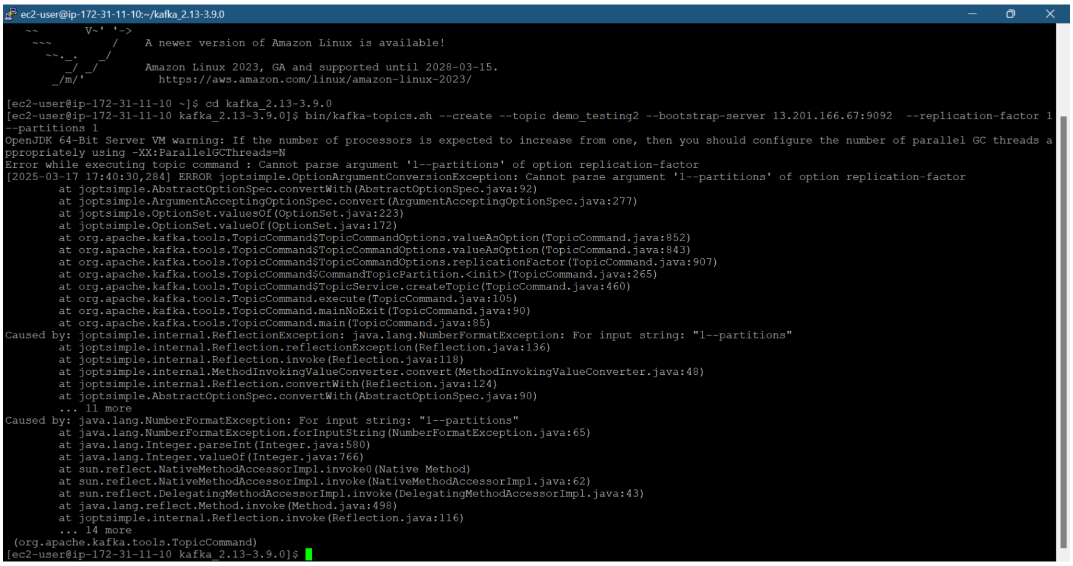
    - sudo chown -R prometheus:prometheus /etc/prometheus
- ➢ Configure prometheus.yml:
    - sudo nano /etc/prometheus/prometheus.yml
- ➢ Create a systemd service for Prometheus:
    - sudo nano /etc/systemd/system/prometheus.service
- ➢ Reload systemd and start Prometheus:
    - sudo systemctl daemon-reload
    - sudo systemctl start prometheus
    - sudo systemctl enable prometheus
- ➢ Check if Prometheus is running:
    - sudo systemctl status prometheus


Prometheus
Query
Alerts
Status
Enter expression (press Shift+Enter for newlines)
Table
Graph
Explain
Evaluation time
No data queried yet
Add query
ubuntu@ip-172-31-15-219: /usr/local/bin/prometheus

```
sabled." component=web http2=false address=[::]:9090
time=2025-03-12T23:46:21.186Z level=INFO source=head.go:795 msg="WAL segment loa
ded" component=tsdb segment=0 maxSegment=0
time=2025-03-12T23:46:21.186Z level=INFO source=head.go:832 msg="WAL replay comp
leted" component=tsdb checkpoint_replay_duration=23.921µs wal_replay_duration=3.
602525ms wbl_replay_duration=617ns chunk_snapshot_load_duration=0s mmap_chunk_re
play_duration=1.518µs total_replay_duration=3.7354ms
time=2025-03-12T23:46:21.190Z level=INFO source=main.go:1249 msg="filesystem inf
ormation" fs_type=EXT4_SUPER_MAGIC
time=2025-03-12T23:46:21.190Z level=INFO source=main.go:1252 msg="TSDB started"
time=2025-03-12T23:46:21.190Z level=INFO source=main.go:1437 msg="Loading config
uration file" filename=prometheus.yml
time=2025-03-12T23:46:21.194Z level=INFO source=main.go:1476 msg="updated GOGC"
old=100 new=75
time=2025-03-12T23:46:21.194Z level=INFO source=main.go:1486 msg="Completed load
ing of configuration file" db_storage=1.214µs remote_storage=2.559µs web_handler
=799ns query_engine=1.203µs scrape=3.264235ms scrape_sd=19.036µs notify=83.346µs
 notify_sd=10.32µs rules=1.517µs tracing=4.173µs filename=prometheus.yml totalDu
ration=3.715761ms
time=2025-03-12T23:46:21.194Z level=INFO source=main.go:1213 msg="Server is read
y to receive web requests."
time=2025-03-12T23:46:21.194Z level=INFO source=manager.go:175 msg="Starting rul
e manager..." component="rule manager"
```

Problems Faced(Research) :

1. **Problem in creating kafka topic**



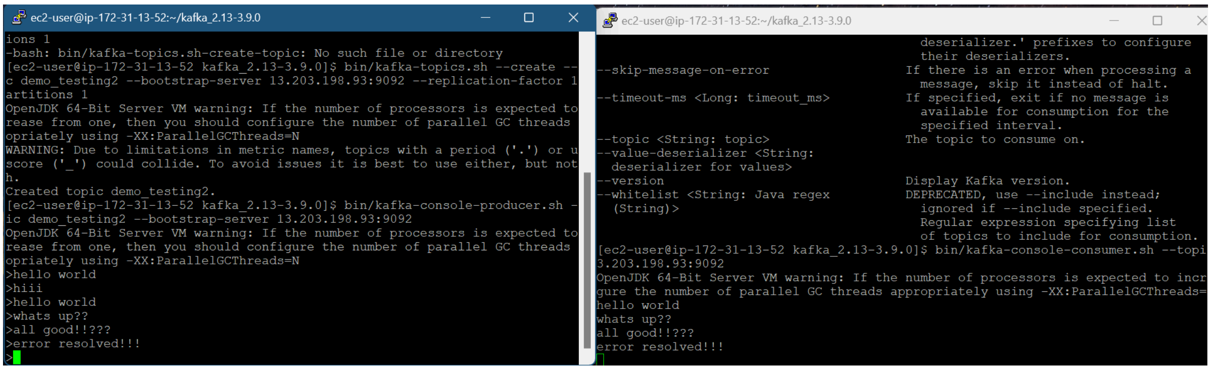
```
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-11-10 ~]$ cd kafka_2.13-3.9.0
[ec2-user@ip-172-31-11-10 kafka_2.13-3.9.0]$ bin/kafka-topics.sh --create --topic demo_testing2 --bootstrap-server 13.201.166.67:9092  --replication-factor 1
--partitions 1
OpenJDK 64-Bit Server VM warning: If the number of processors is expected to increase from one, then you should configure the number of parallel GC threads a
ppropriately using -XX:ParallelGCThreads=N
Error while executing topic command : Cannot parse argument '1--partitions' of option replication-factor
[2025-03-17 17:40:30,284] ERROR joptsimple.OptionArgumentConversionException: Cannot parse argument '1--partitions' of option replication-factor
        at joptsimple.AbstractOptionSpec.convertWith(AbstractOptionSpec.java:92)
        at joptsimple.ArgumentAcceptingOptionSpec.convert(ArgumentAcceptingOptionSpec.java:277)
        at joptsimple.OptionSet.valuesOf(OptionSet.java:223)
        at joptsimple.OptionSet.valueOf(OptionSet.java:172)
        at org.apache.kafka.tools.TopicCommand$TopicCommandOptions.valueAsOption(TopicCommand.java:852)
        at org.apache.kafka.tools.TopicCommand$TopicCommandOptions.valueAsOption(TopicCommand.java:843)
        at org.apache.kafka.tools.TopicCommand$TopicCommandOptions.replicationFactor(TopicCommand.java:907)
        at org.apache.kafka.tools.TopicCommand$CommandTopicPartition.<init>(TopicCommand.java:265)
        at org.apache.kafka.tools.TopicCommand$TopicService.createTopic(TopicCommand.java:460)
        at org.apache.kafka.tools.TopicCommand.execute(TopicCommand.java:105)
        at org.apache.kafka.tools.TopicCommand.mainNoExit(TopicCommand.java:90)
        at org.apache.kafka.tools.TopicCommand.main(TopicCommand.java:85)
Caused by: joptsimple.internal.ReflectionException: java.lang.NumberFormatException: For input string: "1--partitions"
        at joptsimple.internal.Reflection.reflectionException(Reflection.java:136)
        at joptsimple.internal.Reflection.invoke(Reflection.java:118)
        at joptsimple.internal.MethodInvokingValueConverter.convert(MethodInvokingValueConverter.java:48)
        at joptsimple.internal.Reflection.convertWith(Reflection.java:124)
        at joptsimple.AbstractOptionSpec.convertWith(AbstractOptionSpec.java:90)
        ... 11 more
Caused by: java.lang.NumberFormatException: For input string: "1--partitions"
        at java.lang.NumberFormatException.forInputString(NumberFormatException.java:65)
        at java.lang.Integer.parseInt(Integer.java:580)
        at java.lang.Integer.valueOf(Integer.java:766)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at joptsimple.internal.Reflection.invoke(Reflection.java:116)
        ... 14 more
 (org.apache.kafka.tools.TopicCommand)
[ec2-user@ip-172-31-11-10 kafka_2.13-3.9.0]$
```

Troubleshoot:

Kafka topic creation issue was encountered because of insufficient memory. Therefore, we fixed it by adding swap space. Since Kafka needs more memory, we created a 2GB swap file:

- ➢ Create a 2GB swap file:
  - ○ sudo fallocate -l 2G /swapfile
- ➢ Set the correct permissions:
  - ○ sudo chmod 600 /swapfile
- ➢ Set up the swap space:
  - ○ sudo mkswap /swapfile
- ➢ Enable the swap:
  - ○ sudo swapon /swapfile

Output achieved after troubleshoot:


ec2-user@ip-172-31-13-52:~/kafka_2.13-3.9.0
ec2-user@ip-172-31-13-52:~/kafka_2.13-3.9.0

```
ions 1
-bash: bin/kafka-topics.sh-create-topic: No such file or directory
[ec2-user@ip-172-31-13-52 kafka_2.13-3.9.0]$ bin/kafka-topics.sh --create --
c demo_testing2 --bootstrap-server 13.203.198.93:9092 --replication-factor 1
artitions 1
OpenJDK 64-Bit Server VM warning: If the number of processors is expected to
rease from one, then you should configure the number of parallel GC threads
opriately using -XX:ParallelGCThreads=N
WARNING: Due to limitations in metric names, topics with a period ('.') or u
score ('_') could collide. To avoid issues it is best to use either, but not
h.
Created topic demo_testing2.
[ec2-user@ip-172-31-13-52 kafka_2.13-3.9.0]$ bin/kafka-console-producer.sh -
ic demo_testing2 --bootstrap-server 13.203.198.93:9092
OpenJDK 64-Bit Server VM warning: If the number of processors is expected to
rease from one, then you should configure the number of parallel GC threads
opriately using -XX:ParallelGCThreads=N
>hello world
>hiii
>hello world
>whats up??
>all good!!???
>error resolved!!!
>
```

```
                                         deserializer.' prefixes to configure
                                           their deserializers.
--skip-message-on-error                  If there is an error when processing a
                                           message, skip it instead of halt.
--timeout-ms <Long: timeout_ms>          If specified, exit if no message is
                                           available for consumption for the
                                           specified interval.
--topic <String: topic>                  The topic to consume on.
--value-deserializer <String:
  deserializer for values>
--version                                Display Kafka version.
--whitelist <String: Java regex          DEPRECATED, use --include instead;
  (String)>                                ignored if --include specified.
                                           Regular expression specifying list
                                           of topics to include for consumption.
[ec2-user@ip-172-31-13-52 kafka_2.13-3.9.0]$ bin/kafka-console-consumer.sh --topi
3.203.198.93:9092
OpenJDK 64-Bit Server VM warning: If the number of processors is expected to incr
gure the number of parallel GC threads appropriately using -XX:ParallelGCThreads=
hello world
whats up??
all good!!???
error resolved!!!
```