RDS MySQL Engine Slowness Troubleshooting Runbook

This runbook will guide you through the process of troubleshooting RDS MySQL engine slowness using AWS CLI commands. We'll cover steps to check performance metrics, database status, and other configurations that might be causing the slowness.

Step 1: Identify the Affected RDS Instance

- If the RDS instance identifier is already known, proceed to Step 2. Otherwise, use the AWS Management Console or AWS CLI to identify the affected RDS instance exhibiting slowness.

Step 2: Check RDS Instance Status Use the following AWS CLI command to check the status of the RDS instance:

bash

Copy code

```
aws rds describe-db-instances --db-instance-identifier YOUR_DB_INSTANCE_IDENTIFIER
```

Ensure that the instance is in the "available" state and has not reported any recent issues or events.

Step 3: Review RDS Instance Performance Metrics

- To check CPU utilization and other important performance metrics for the RDS instance, use the following command:

bash

Copy code

```
aws cloudwatch get-metric-statistics --namespace "AWS/RDS" --metric-name
"CPUUtilization" --dimensions
"Name=DBInstanceIdentifier,Value=YOUR_DB_INSTANCE_IDENTIFIER" --start-time
YOUR_START_TIME --end-time YOUR_END_TIME --period 300 --statistics Average
```

- Additionally, review other key metrics like "DatabaseConnections," "FreeableMemory," and "ReadLatency" to identify potential bottlenecks.

Step 4: Enable Enhanced Monitoring (Optional) If you haven't already, consider enabling Enhanced Monitoring for the RDS instance to get more detailed OS-level metrics. This can provide better insights into system resource usage.

Step 5: Check Database Load

- Use the following command to check the current connections and queries running on the RDS instance:

bash

Copy code

```
aws rds describe-db-connections --db-instance-identifier
YOUR_DB_INSTANCE_IDENTIFIER

aws rds describe-db-engine-versions --db-parameter-group-family mysql8.0 --query
"DBEngineVersions[0].DBParameterGroupFamily"
aws rds describe-db-instance-performance --db-instance-identifier
YOUR_DB_INSTANCE_IDENTIFIER --metric-name Queries
```

- Analyze the output to identify any long-running or slow queries that may be impacting performance.

Step 6: Check Database Logs Review the RDS instance's error log and slow query log to identify any database-related issues or queries causing slowness.

bash

Copy code

```
aws rds download-db-log-file-portion --db-instance-identifier
YOUR_DB_INSTANCE_IDENTIFIER --log-file-name error/mysql-error.log --starting-token
0
aws rds download-db-log-file-portion --db-instance-identifier
YOUR_DB_INSTANCE_IDENTIFIER --log-file-name slowquery/mysql-slowquery.log
--starting-token 0
```

Step 7: Analyze Storage Performance Check if there are any storage-related issues impacting performance.

bash

Copy code

```
aws cloudwatch get-metric-statistics --namespace "AWS/RDS" --metric-name
"BurstBalance" --dimensions
"Name=DBInstanceIdentifier,Value=YOUR_DB_INSTANCE_IDENTIFIER" --start-time
YOUR_START_TIME --end-time YOUR_END_TIME --period 300 --statistics Average
```

Step 8: Check Parameter Group Settings Review the RDS instance's parameter group settings to ensure they are optimized for your workload.

bash

Copy code

```
aws rds describe-db-parameters --db-parameter-group-name
YOUR_DB_PARAMETER_GROUP_NAME
```

Step 9: Review Security Group Rules Ensure that the RDS instance's security group rules are correctly configured to allow necessary traffic.

bash

Copy code

```
aws rds describe-db-instances --db-instance-identifier YOUR_DB_INSTANCE_IDENTIFIER
--query "DBInstances[*].VpcSecurityGroups[*].VpcSecurityGroupId"
aws ec2 describe-security-groups --group-ids YOUR_SECURITY_GROUP_ID
```

Step 10: Check Network Connectivity Verify that there are no network-related issues affecting the RDS instance's connectivity.

Step 11: Review Instance Class and Storage Ensure that the RDS instance class and storage capacity are appropriate for your workload.

Step 12: Check AWS Support (If Required) If the issue persists or is beyond your expertise, consider reaching out to AWS Support for further assistance.