CPU throttling or consumption

Last updated by | Lisa Liu | Nov 6, 2020 at 10:35 AM PST

Issue – CPU throttling with no connections nor workload

- Customer might report that they are seeing high CPU consumption on database server than usual.
- Customer could report that CPU is throttling even though workload is the same or null and want to know what is consuming the resource.

Investigation/Analysis

To understand the issue, there are a couple of details to get from customer like:

- When they start seeing CPU consumption increases for some unknown reason. Exact timeframe or frequency is important to know.
- ASC should help see and compare the workload reviewing IOPs and number of connections, including if backup was running at that time.
- If for no reason CPU keep throttling frequently, then you should consider investigate the server has or had replica in the past, and check active replication slots that might have left over after replication was broken. You can use this Kusto Query to check if there are any possible active replication slots. Mitigation
- A connection pooler is good recommendation if customer is having a lot of short live connections this can be check using this Kusto Query.
- Server IO capacity, if the server using it capacity at more than 90%, the CPU will show a lot of pressure and therefore become slow, recommendation in this will asking customer to increase storage or scale out number vCores.
- For server CPU throttling for active left-over replication slots, customer might want to find the active slots by running this query “SELECT * from pg_replication_slots;” and delete using this one “SELECT pg_drop_replication_slot('Slotname');”.

Use the following Query to check the replication slots still active:

MonDmPgSqlReplicationStatsPrimary

| where LogicalServerName == "[server_name]"

| project LogicalServerName, pid, slot_name, slot_type, active, application_name, wal_sender_state, restart_lsn, backend_start, PreciseTimeStamp, backend_xmin, total_lag_in_bytes, sync_state, current_wal_lsn, sent_lsn, write_lsn, flush_lsn, replay_lsn, sync_priority, ClusterName, NodeName

| order by PreciseTimeStamp desc

Public Doc Reference (optional)

- https://docs.microsoft.com/en-us/azure/postgresql/concepts-logical#how-to-drop-a-slot

Internal Reference (optional)

- https://servicedesk.microsoft.com/#/customer/cases?caseNumber=120081224006535

Root Cause Classification

Root Cause: Azure Open Source DB V2\Performance\User Issue/Error\VCore Limit\CPU

How good have you found this content?