Create or prmote a replica taking longer time to complete

Last updated by | Hamza Aqel | Aug 2, 2022 at 9:01 AM PDT

Sometimes creating or promoting a replica might take longer time than the customer expects, or might fail for some reason, use the below troubleshooting steps to track the replica creation:

1- Identify the Request_ID :

MonManagement

| where TIMESTAMP >= datetime('StartTime') and TIMESTAMP <= datetime('EndTime')

| where operation_parameters contains "<PartnerElasticServerName>" and operation_parameters contains 'pg-replica-servername'

| project request_id

request_id
51EA5C84-88F1-4361-9850-F34554281FCA

2- Check the request_id details to see if there is any error there:

MonManagement

| where request_id == "Request_ID"

| project originalEventTimestamp, operation_parameters , NodeName, transaction_id, context_instance_name, event, state_machine_type, state, old_state, new_state, action, fsm_event, exception_type, message, error_code,

error_message, stack_trace, keys, elapsed_time

3- If no errors there , the server is still creating and trying to recover from the Primary , you can check from the Sandbox to confirm :

MonRdmsPgSqlSandbox

| where TIMESTAMP >= datetime('StartTime')

| where LogicalServerName == "pg-replica-servername"

| project TIMESTAMP,text

TIMESTAMP	text
2022-03-14 06:48:03.8731368	2022-03-14 06:48:00 UTC-622edc5d.2c-LOG: recovering 0000000100000511000000DC
2022-03-14 06:48:03.8731567	2022-03-14 06:48:02 UTC-622edc5d.2c-LOG: recovering 0000000100000511000000DD
2022-03-14 06:48:08.8724349	2022-03-14 06:48:05 UTC-622edc5d.2c-LOG: recovering 0000000100000511000000DE
2022-03-14 06:48:08.8724543	2022-03-14 06:48:08 UTC-622edc5d.2c-LOG: recovering 0000000100000511000000DF

So the customer in this case should wait till the recovery process is complete.

After the server is created, it is expected that we will see a replication lag, so the replica will start to sync with the primary and you can monitor this by the below:

MonDmPgSqlReplicationStatsPrimary

| where TIMESTAMP >= datetime('StartTime')

| where LogicalServerName == "pg-master-servername"

| project LogicalServerName, slot_name, active, slot_type, wal_sender_state, PreciseTimeStamp, application_name, total_lag_in_bytes, receiving_lag_in_bytes, replaying_lag_in_bytes, current_wal_lsn, sent_lsn, restart_lsn, sending_lag_in_bytes, write_lsn, write_lag, flush_lsn, flush_lag, replay_lsn, replay_lag

| order by PreciseTimeStamp desc

To get the full details for the above , you can refer to this Replication TSG

Promote a Replica:

For Promoting replica cases you can use the same previous steps, taking into consideration for the replica promote under lag scenario, the replica promote will taking more time to catch up with the primary before it become a standalone server , to get a full details about the replica lag you can refer to this Replication TSG.

1- Below query will show if replica is promoting state in the backend. In this case you can see if is still waiting. If promote is complete, you will see a confirmation message at the end of this query result

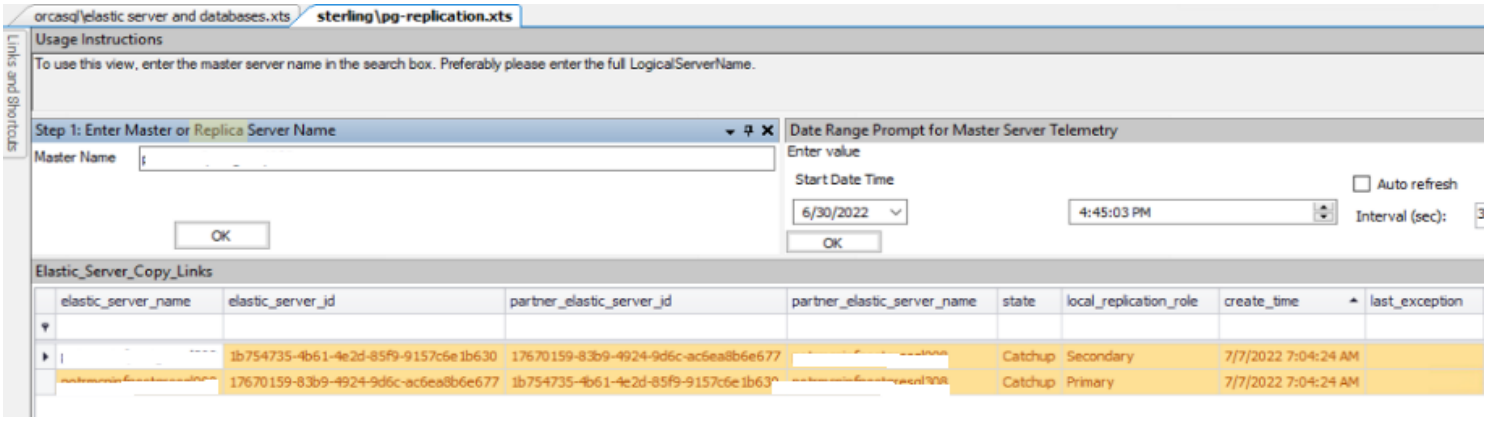
```
let ServerName = "pgservername";
let StartTime = ago(3d);
let EndTime = ago(1m);
MonRdmsInstanceAgent
| where originalEventTimestamp >= StartTime and originalEventTimestamp <= EndTime
| where LogicalServerName == ServerName
| where message_systemmetadata contains "promote"
| project originalEventTimestamp, LogicalServerName, process_id, NodeName, message_systemmetadata| order by or
```

originalEventTimestamp	LogicalServerName	message_systemmetadata
2022-07-14 05:30:28.3296977	pgservername	[PostgresAgentActor].PromoteReplicaServer: Promoting the server
2022-07-14 05:30:28.3351052	pgservername	[PostgresAgentActor].PromoteReplicaServer: Successfully established connection to the server
2022-07-14 05:30:28.3351512	pgservername	[FabricHelper].GetConfigSetting: Retrieved config setting 'EnablePromoteReplicaImmediately' from package SQL.Config, section FeatureSwitches as: [off]
2022-07-14 05:30:28.3455897	pgservername	[PostgresAgentActor].PromoteReplicaServer: Creating promoteTriggerFile=\\wasd2prodcnno2apfe98.file.core.chinacloudapi.cn\1b7547354b614e2d85f99157c
2022-07-14 05:30:28.3592203	pgservername	[PostgresAgentActor].PromoteReplicaServer: Successfully created promoteTriggerFile=\\wasd2prodcnno2apfe98.file.core.chinacloudapi.cn\1b7547354b614e2
2022-07-14 05:30:28.4794990	pgservername	[PostgresAgentActor].IsServerPromoted: Checking if server is promoted
2022-07-14 05:30:28.4805971	pgservername	[PostgresAgentActor].IsServerPromoted: recoveryDoneFilePath=\\wasd2prodcnno2apfe98.file.core.chinacloudapi.cn\1b7547354b614e2d85f99157c6e1b630\D
2022-07-14 05:30:38.5581312	pgservername	[PostgresAgentActor].IsServerPromoted: Checking if server is promoted
2022-07-14 05:30:38.5582399	pgservername	[PostgresAgentActor].IsServerPromoted: recoveryDoneFilePath=\\wasd2prodcnno2apfe98.file.core.chinacloudapi.cn\1b7547354b614e2d85f99157c6e1b630\D
2022-07-14 05:30:48.6258287	pgservername	[PostgresAgentActor].IsServerPromoted: Checking if server is promoted

2- Open "elastic servers and databases" view and look up the replica status:

state	create_time	elastic_server_create_mode	customer_subscription_id	resource_group	elastic_server_type	elastic_server_edition	is_pfs	elastic_serv
WaitingForStopReplication	7/7/2022 7:04:35 AM	GeoSecondary	ea58df11-3f3e-4efc-8adc-2f27b4092480	POTRMCNINFRESGP014	PostgreSQL.Server.PAL	GeneralPurpose	1	1b754735-4

3- Open pg-replication view in the replica region and look up the replica. Check the states of the link row. You will see two rows if same region replica, or otherwise only one row for cross region replica



If the replica is in "WaitingForStopReplication" state in elastic_servers, it means it is in the process of being promoted. It can be in this state for long time, if replica is promoted when it has a lot of lag, keep moniotring the prmote,and lag status if it is moving forward after sometime, in case of stuck operation and no progress, feel free to file an ICM.