AIQ Platform Failure Scenarios

This section covers the below broader AIQ Non-PCF platform level or component level failure scenarios.

- AWS Region becomes Unavailable
- One CRDB EC2 becomes unavailable in One region in CRDB Cluster.
- Multiple/All CRDB EC2 In one region becomes unavailable/unreachable in CRDB Cluster.
- One Hazelcast EC2 becomes unavailable in One region .
- Multiple/All Hazelcast EC2 In one region becomes unavailable/unrechable in Cluster.
- Similar to above track scenarios when One or Multiple/All of EC2's in a region are unavailable for Squid/Logstash/Nifi Components.
- Troubleshooting Procedures for PCF To AIQ Backend Endpoints Connectivity issues.
- Troubleshooting Procedures for CRDB Response Time Spikes Causing Timeouts on consuming AIQ Services.
- Teletran platform Unavailable in one region.
- AIQ Logs Not showing up in Teletran in One/Both Regions.

Serial no#	Component	Reference link
1	AWS region	AWS Region Unavailable Scenario
2	CRDB	AIQ Production - CRDB Failure Scenarios
3	CRDB	Troubleshooting Procedures - CRDB Cluster Response Time Spikes Causing AIQ Service Timeouts
4	Hazelcast	AIQ Production - Hazelcast Failure Scenarios
5	Logstash	AIQ Production - Logstash Failure Scenarios
6	Cloudfoundry	Troubleshooting Procedures - PCF To AIQ Backend Endpoints Connectivity Issues
7	Teletraan	AIQ Production - Logs Missing In Teletraan In One Region

For each of the above AIQ Platform Failure Scenarios, include the below sections (as a standard template), wherever applicable :

Standard template

```
1. Impact :
   a. Single Node Failure:
   b. Multiple Node Failure:
   c. Entire cluster is down in a specific Region:
2. Possible Cause Of Failure
   a. Possible causes for Single/Multiple Nodes down in a specific region:
   b. Possible causes for the Entire cluster is down in a specific region:
3. Troubleshooting steps
   a. Single/Multiple nodes down:
   b. Entire cluster down in a specific region:
4. Post troubleshooting validation
5. Communication
```

AIQ Production - CRDB Failure Scenarios

The following scenarios explain when one/Multiple CRDB nodes are down in a specific region or an entire CRDB cluster is down in a specific region.

Impact :

Single Node Failure: There is no impact on the applications upon failure of a single cockroach node in a cluster as all the nodes are grouped under an ELB and load would be distributed equally within other nodes in a cluster.

Multiple Node Failure: There is no impact on the applications upon failure of one or more cockroach node in a cluster as all the nodes are grouped under an ELB and load would be distributed equally within other nodes in a cluster.

Entire CRDB cluster is down in a specific Region: All applications accessing a cockroach cluster in the respective region will experience complete failover.

Possible causes for Single/Multiple Nodes down in a specific region :

- degradation of the underlying hardware for respective cockroach ec2 instance results in the complete shutdown.
- cockroach process on steel cloud VM's in central PDC nodes can go down due to clock sync exceptions
- The entire availability zone with corresponding cockroach nodes residing is down.

Possible causes for the Entire CRDB cluster is down in a specific region

- The VPC assigned to the corresponding cockroach cluster might be down potentially leading to a broader issue within AWS.
- Direct connect connection issue between Comcast and AWS

Troubleshooting steps :

Single/Multiple CRDB nodes down :

- login to the CRDB production Admin console using https://internal-aiqcrdb-ss-prd-elb-823876589.us-east-1.elb.amazonaws.com:8080/#/overview/list and check the count for the number of live nodes and verify if there are any dead/suspected nodes.

- if the user is unable to login to the affected cockroach ec2 instance due to the underlying hardware issues, log in to the AWS console https://awslogin.comcast.com.
  Remove the affected cockroach ec2 instance from the respective ELB, and stop and start the ec2 instance.
- if the user is able to log in, Identify the respective dead node from the console and login to the respective node and check for cockroach node logs under the path **/root/cockroach/cockroach-data/logs/cockroach.log** for connectivity issues.
- next steps if there is no useful info from the logs regarding the issue, reach out to cockroach enterprise support via slack channel **cockroachlabs.slack.com**

- cases in which the CRDB process goes down on the steel cloud VM's in the central PDC nodes, cron jobs are in place to run the NTP scripts in the background and once the underlying Vmotion is recovered the CRDB process is up and running .

- For Mulitple node failures, identify the respective region with respect to the affected cockroach nodes and flip the traffic away from the region.
- If multiple nodes are failing due to the underlying hardware failures in EC2 instances, follow the same troubleshooting steps followed for single CRDB node failure.
- log in to the respective CRDB nodes in the region where traffic is active and check for logs with respect to the failed nodes under the path **/root /cockroach/cockroach-data/logs/cockroach.log**
  For instance, if Multiple CRDB nodes are failing in the east, switch the traffic to the west and check for logs under west nodes regarding connectivity issues with east nodes.
- Check for the #pmr_war_room and #aws slack channel for any on-going broader issue within the availability zone and create a support ticket with the AWS support to troubleshoot the issue further as login **https://awslogin.comcast.com. < support < support center < create** case if required
- once the issue is resolved flip traffic back to Active-Active

Entire CRB cluster down in a specific region

- login to the CRDB production Admin console using https://internal-aiqcrdb-ss-prd-elb-823876589.us-east-1.elb.amazonaws.com:8080/#/overview/list and check the count for the number of live nodes and verify if there are any dead/suspected nodes.

- identify the respective region with respect to the affected cockroach nodes and flip the traffic away from the region.
- check the #pmr_war_room slack channel for any on-going broader issue regarding the AWS direct connect.
- reach out to L1 support via #re_ea_l1_ccc slack channel co-ordinate and create a sev1 incident bridge.
- reach out to AWS enterprise support and create a support ticket for further troubleshooting.
- once the issue is resolved, flip the traffic back to Active-Active.

Post troubleshooting validation

- Check the cockroach production admin console https://internal-aiqcrdb-ss-prd-elb-823876589.us-east-1.elb.amazonaws.com:8080/#/overview/list and verify Node status under live nodes section and make sure there are no suspected or dead nodes.
- log in to the respective cockroach node and check the status of the cockroach process using the command **systemctl status cockroachdb**

Communication

- Acknowledge the issue initially and post updates in **#aiq-convoy slack channel.**
- Post updates in **#aiq-convoy** every 30 min regarding any background of the issue and troubleshooting steps.
- Post final round of updates with RCA or summary regarding the issue in **#aiq-convoy slack** once the issue is resolved.

Comcast Confidential - Any unauthorized disclosure or use is strictly prohibited

AIQ Production - Hazelcast Failure Scenarios

This scenario is about Hazelcast Database in Production environment fails.

Impact:

a) Single Node failure within an AZ:

There shall be no impact to the application as the nodes are spread across Availability zones/ subnets, in a region. Three nodes in each AZ across two AZs

b) Multiple nodes failure within an AZ:

There is no significant impact to the application as backup partitions are available on another AZ/subnet

c) Entire region nodes failure:

This failure is "**critical**"

Possible Causes:

a) Single node down in an AZ:

- Heavy CPU usage and awaiting cycles
- Scheduled AWS retirement on EC2
- Service is down
- Network connectivity
- Underlying EC2 maintenance

b) Multiple nodes down in an AZ:

- AWS data center network outage
- Security group changes
- Configuration change
- Networking lapse
- Hardware failures

c) Entire region nodes failure:

- AWS data center outage
- Network/hardware outages

Troubleshooting Steps:

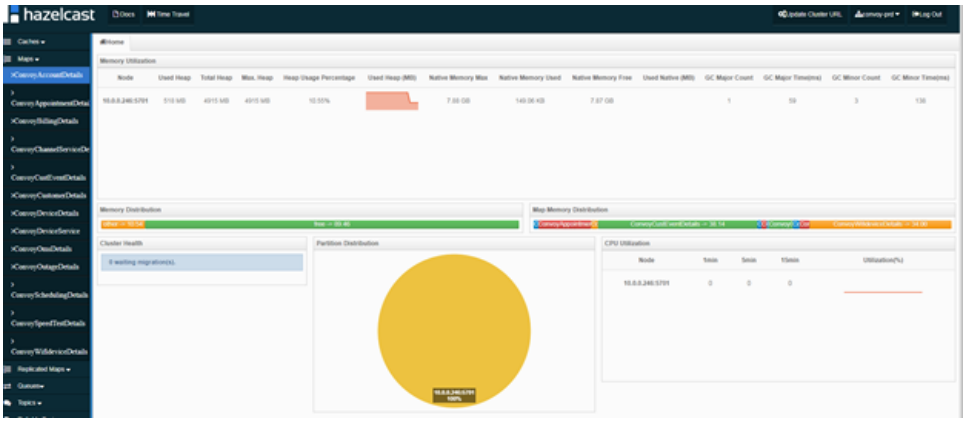
Step-by-step guide

Monitoring and other troubleshooting

One AIQ AWS Region based node becomes Unavailable/Unreachable :

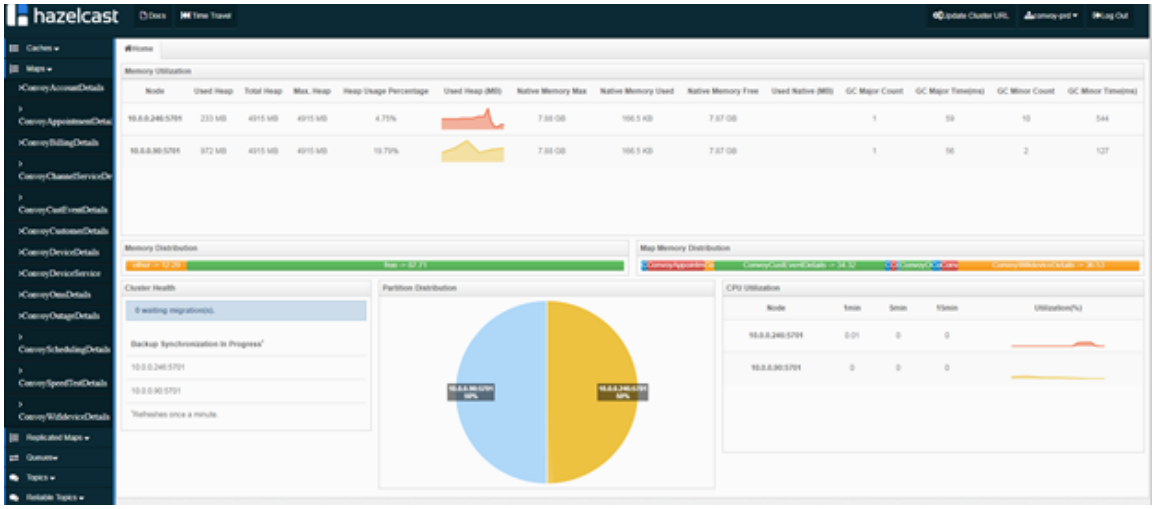
When an AIQ AWS Region based Cluster node in a region becomes Unavailable/Unreachable, typically all the maps and partitions are moved from the dead node to healthy node. It can be observed on Management UI. Login to UI with Credentials

1. Log on to the Management Center Console,to verify the health of Cluster and nodes, with migration status.
   When the node is down, you shall see only the remaining nodes in the cluster, containing Maps. In this scenario of two node cluster, only one node is reported, as below

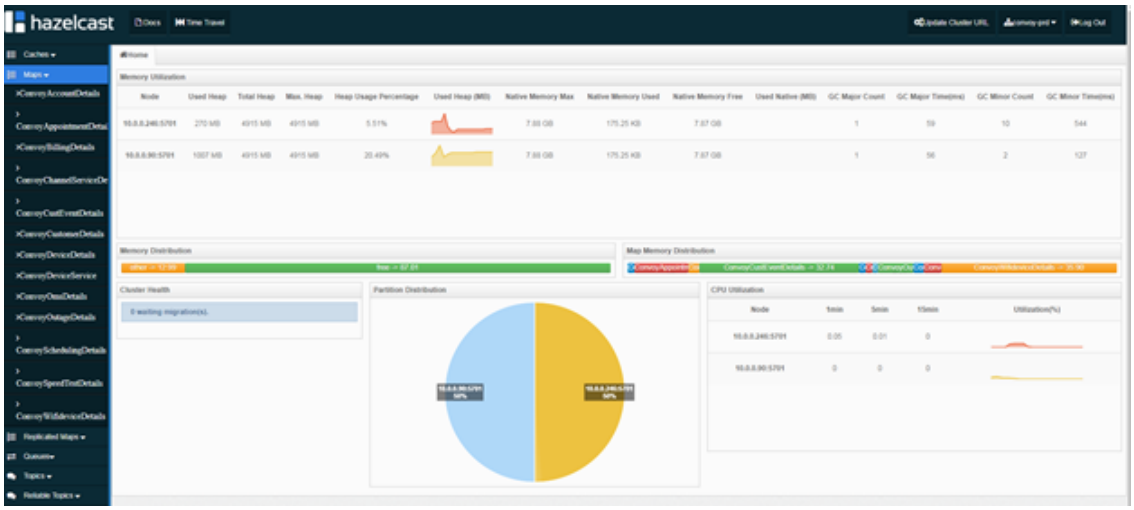


2. When you observe "no waiting migrations" in Cluster Health and 100% in Partition Distribution pie chart, it is evident that the single node is containing all the partitions of the cluster.

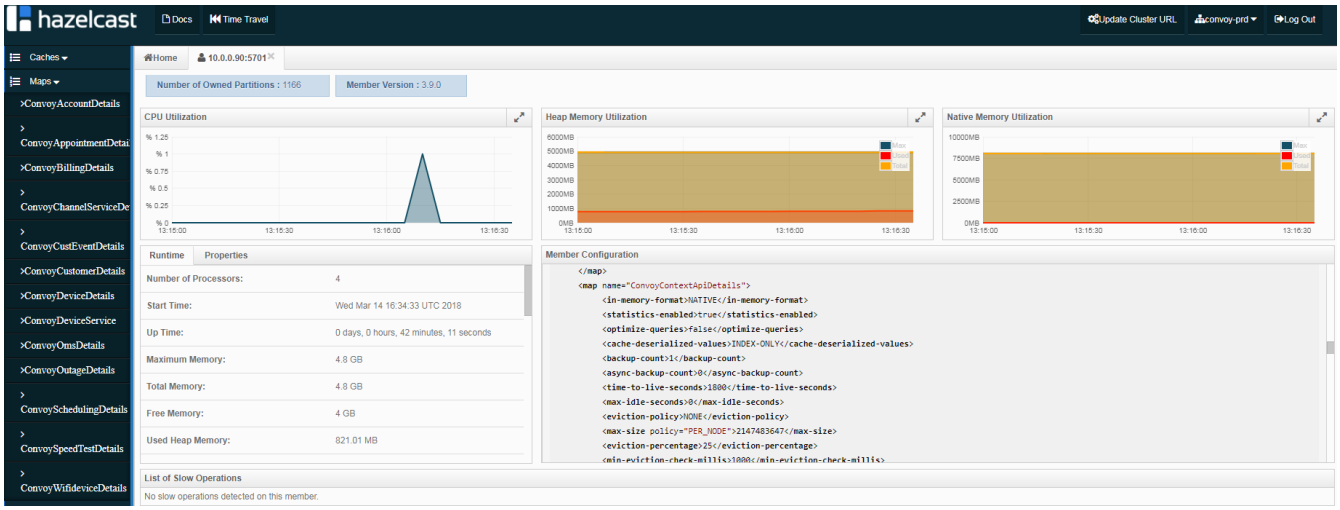
3. Though the Partitions are redistributed and no waiting migrations, Backup synchronization is in Progress.



4. The final stage after Backup synchronization is complete



5. We can validate new Map created as part of Console configuration, while selecting the member node and scroll in member configuration.



One AIQ AWS Region becomes Unavailable/Unreachable :

There are two scenarios when the entire AWS region is unavailable for Hazelcast cluster.

a) When there is network related issue

b) When there is an underlying hardware issue

In either of the cases, we move the traffic to other end of the country's AWS region, using GSLB based Album traffic flips. This helps us with minimal damage or impact.

Network related issue does NOT contribute any data loss in Hazelcast cluster, in the region affected. There are no procedures further to perform.

In case of Hardware related issue, if all nodes are down, then there shall be a total data LOSS. The entire cache hosted on memory gets deleted and no recovery possible from that cluster. As part of troubleshooting the impacted region nodes, it is important to validate all key resources like Cache and CPU. When the impacted cluster is functional and ready to be added, after troubleshooting, there is an Important additional step to perform.

A Hazelcast Map WAN Sync operation, which includes synchronization maps and partitions of Available region cluster to Impacted region cluster. This ensures the data is approximately same(since the transitional data transfer is not synced). A significant increase in performance and capacity issues for respective backends is expected, which are called to rebuild Cache, shall see a spike in traffic. Refer AIQ Hazelcast Map WAN Sync Procedure. Upon completion of Sync, the traffic can be balanced across both regions, using GSLB based load-balancers.

At that point onwards, we remain as "Active-Active"

Network ports check

```
netstat -anv | grep :8380*
```

Check free memory

```
free -m
```

Remediation steps:

Post troubleshooting validation steps

1. Monitor Management UI for clustered node health and status
2. Check Appdynamics for backends response times and connectivity

Communication

- **# aiq-va-ops**
- **# aiq-all**
- **# pmr_war_room**

- **# re_ea_l1_ccc**

Related articles

Hazelcast Map Creation and Rolling restart

Hazelcast operations

AIQ Hazelcast Map WAN Sync Procedure

AIQ Individual Components Restart Procedures

AIQ Traffic Flip Procedures

AIQ Production - Logs Missing In Teletraan In One Region

This scenario is about application, transactional, websdk logs for Production environment in Teletraan cluster. Teletraan team manages infrastructure for aiQ

Impact:

1) Troubleshooting Failback errors and other Kibana related Elastalerts

2) Application visibility lapse over data metrics in a region

Possible Causes:

1) Disk usage beyond permissible limit on Teletraan cluster nodes and Cluster RED

2) Elasticsearch endpoints does not respond for connectivity from Logstash

3) Security group changes on either Logstash or Teletraan end

4) Firewall connectivity/VPC peering related issues

5) Direct Connect/ AWS outage for network

Troubleshooting Steps:

Step-by-step guide

Logs missing in ONE region:

1. Check if there is an ongoing outage on Datacenter in AWS and/or Comcast Direct-connect region, in "**#pwr_war_room**" and "**#aws**" slack channels, to rule out external dependencies
2. Check the logstash production ELB for any reported failed nodes or bad nodes. Look through network monitoring on Cloudwatch metrics at ELB level for the region affected.
3. Login through SSH with 'aiq-prod' key-pair onto one of the instance under the ELB. Switch to root user for administration
4. Under the logstash logs path "/var/log/logstash", look for current day log file and tail it for errors
5. Skim through logs for ERROR and WARN entries. Repeat the same steps on other instances under the ELB
6. If there is a reported Elasticsearch cluster non-availability reason, reach out to "**#teletraan_support**" slack channel and report the status on "**#aiq-va-ops**" channel with an error information
7. If there are intermittent connections and/or port is intermittently sending the logs to end points, check the CPU usage and Free memory available on instance. It could be an overhead on resources and waiting for CPU cycles.

Monitoring and other troubleshooting

1. Check Kibana for logs based on either "aiq-concise-*", "aiq-websdk-*", "aiq-syslog-*" indices' patterns. Filter the time-series histogram with "concise-datacenter" to identify Region based metrics.
2. To obtain Operating system/CPU/memory metrics on logstash instances, use Appdynamics metrics browser. Skim through usage under linux plugin for respective tiers and nodes
3. Cloudwatch shall help with the network level metrics of ELB and instances.

Commands handy

Network ports check

```
netstat -anv | grep :54*
```

Check free memory

```
free -m
```

Tail logs for logstash
`tail -f /var/log/logstash/logstash-plain.log`

Remediation steps:

1. Though Logstash is not available in one region, our setup maintains **cross-region based logging**. So, at any given point of time, you shall have logs pushed to both the region based Teletraan clusters concurrently.
2. Kibana on the other region shall contain the logs pertinent to datacenter. However, there could be a latency factor, since they are in-time writes across country
3. Engage appropriate teams and stakeholders to check for network/security group/firewall policy changes

Please follow the process for login using AWS SAML credentials (aws_adfs_auth) and authenticate yourself for these activities on EC2

Related articles

Logstash

Slack Channels and Alerts for Operational Support

Network ports of Components

- Troubleshooting Procedures - CRDB Cluster Response Time Spikes Causing AIQ Service Timeouts
- AIQ Production - CRDB Failure Scenarios
- Troubleshooting Procedures - PCF To AIQ Backend Endpoints Connectivity Issues
- AIQ Platform Monitoring Info
- AIQ Runbook

Comcast Confidential - Any unauthorized disclosure or use is strictly prohibited

AIQ Production - Logstash Failure Scenarios

Logstash instances are spawn as individual nodes under an ELB. These instances do not have dependencies among them. Remember, logstash is not a cluster and so any transitional logs shall be dropped. The algorithm on logstash is Write-Atleast-Once to Elasticsearch. The following scenarios explain when one/Multiple Logstash nodes are down in a specific region.

Impact :

Single Node Failure: There is no impact on the concise/sys/websdk logs being shipped to Kibana upon failure of a single Logstash node as all the nodes are grouped under an ELB and load would be distributed equally within other nodes in a cluster.

Multiple Node Failure: There is no impact on the concise/sys/websdk logs being shipped to Kibana upon failure of a single Logstash node as all the nodes are grouped under an ELB and load would be distributed equally within other nodes in a cluster.

Possible causes for Single/Multiple Nodes down in a specific region :

1. Spikes in CPU, memory, disk usage, and Network metrics can result in the downtime of concise/sys/websdk services running on the respective logstash node.
2. spikes in the JVM heap memory can result in the downtime of concise/sys/websdk services.
3. degradation of the underlying hardware for respective logstash ec2 instance results in the complete shutdown.

Troubleshooting steps :

Single/Multiple Logstash nodes down :

1. Login into the respective Logstash node and verify if all the 3 services(concise, sys,websdk) are up and running on the logstash node.
2. Check if memory is free on the instance and the heap is not overloaded. There could be a lot of buffered caches piled up and not released to "free" memory. Thus causing a difference between "free" and "available" memory from "free -m" output
3. If the buffered cache is piled up, use this command to seamlessly and on-the-fly execution.

Free buffered memory
`sync; echo 3 > /proc/sys/vm/drop_caches`

If there are any issues with the disk

Free older logs
`find/var/log/logstash/-mtime +14 -name "*.log"-print -delete;`

4. Check load averages and CPU usage on the server. it could impact the server when you observe the highest contenders of CPU usage and kill the process if that is not critical. In doing so, make sure you have Java processes checked before killing them.

Post troubleshooting validation :

Communication:

Comcast Confidential - Any unauthorized disclosure or use is strictly prohibited

AIQ Production - logstash one instance down in a region

Logstash instances are spawn as individual nodes under a ELB. These instances does not have dependencies among them. Remember, logstash is not a cluster and so any transitional logs shall be dropped. The algorithm on logstash is Write-Atleast-Once to Elasticsearch.

Impact:

1) There shall be no logs sent to Teletraan via this node. All transient logs shall be lost and does not appear in Kibana.

2) Clog up all Memory and CPU resources, causing delayed response over SSH connections

Possible Causes:

1) Memory full due to Cached buffers not released

2) Long running process killed and disk/file system full

3) Service configuration changes with potential errors

Troubleshooting steps:

Step-by-step guide

Logstash ONE instance down in a region

1. Check if the instance is running with good disk,CPU, memory and network
2. Pull it out of rotation from ELB and start investigating. While doing so, appd/cloudwatch might trigger alerts based on "unhealthy nodes". So removing the node out of ELB shall help from the alert going off

3. Check if the instance is running Logstash service and the ports are open/listening
4. Skim through logs for any errors and warnings during the timeframe
5. Check Appd for metrics on network cpu and other OS related spikes
6. Check if memory is free on the instance and heap is not overloaded. There could be a lot of buffered cache piled up and not released to "free" memory. Thus causing a difference between "free" and "available" memory from "free -m" output
7. If the buffered cache is piled up, use this command to seamlessly and on-the-fly execution.

Free buffered memory
`sync; echo 3 > /proc/sys/vm/drop_caches`

If there are any issues with disk

Free older logs
`find /var/log/logstash/ -mtime +14 -name "*.log" -print -delete;`

8. Check if there is a EBS volume disk issue and engage with AWS team as needed.
9. If the backend Elasticsearch cluster is not reachable, the logstash shall queue the logs to push and retry once as per algorithm. During this phase, it could pile up a lot of resources and cause service to stop as well
10. Restart of logstash service can help in that scenario
11. If the logging level is set to "INFO", there could be logging at verbose level, causing quicker disk usage. This can also help understand disk full issue.
12. Check load averages and CPU usage on the server. it could impact the server when you observe highest contenders of CPU usage and kill the process if that is not critical. In doing so, make sure you have Java processes checked before killing them

Comcast Confidential - Any unauthorized disclosure or use is strictly prohibited

Related articles

- Troubleshooting Procedures - CRDB Cluster Response Time Spikes Causing AIQ Service Timeouts
- AIQ Production - CRDB Failure Scenarios
- Troubleshooting Procedures - PCF To AIQ Backend Endpoints Connectivity Issues
- AIQ Platform Monitoring Info
- AIQ Runbook

AWS Region Unavailable Scenario

Impact:

All aIQ core components hosted in the particular AWS region will be impacted.

Possible causes :

- Failures in connectivity between AWS direct connect and VPC's hosting our core aIQ components.
- Disruption in AZ in of the AWS region us-east or us-west.

Troubleshooting steps :

- Failures in connectivity between AWS direct connect and VPC's hosting our core aIQ components.

  step 1: Login into the https://awslogin.comcast.com
  step 2: Navigate to the Direct connect service and check the status of the Virtual interfaces are in an available state.
  step 3: if any of the Virtual interfaces are in the unavailable state, Identify the respective unavailable VIF's attached to the respective virtual gateways in specific region US-EAST or US-WEST.
  step 4: Flip aIQ traffic to the respective region US-EAST or US-WEST accordingly.
  step 4: reach out to the cloud sre team via #aws slack channel and troubleshoot if there is any on-going Maintenance going on with the AWS direct connect.
  step 5: check the #pmr_war_room slack channel for updates regarding the issue and reach out to L1 support via #re_ea_l1_ccc slack channel co-ordinate and create a sev1 incident bridge.
  step 6; Validate the issue is resolved and flip traffic back to Active-Active.

- Disruption in AZ in of the AWS region us-east or us-west.

  step 1: check the cloud sre team via #aws slack channel and identify the effected AZ with respect to US-east or US-west and flip traffic away from the region.
  step 2: follow up in the #pmr_war_room slack channel for any updates regarding the on-going issues.
  step 3: once the issue is resolved, flip aIQ traffic back to Active-Active.

Post troubleshooting validation :

- check the cloud sre team via #aws slack channel and validate if the issue is resolved.

Communication :

- Acknowledge the issue initially and post updates in **#aiq-va-ops slack channel.**
- Post updates in **#aiq-va-ops** every 30 min regarding any background of the issue and troubleshooting steps.
- Post final round of updates with RCA or summary regarding the issue in **#aiq-va-ops slack** once the issue is resolved.

CRDB Actions To Perform During Firewall/Network Issues /Maintenances

Why we need to disable leaseholder in CRDB?

During network interruption we have observed CRDB become unstable and unresponsive, to over come the situation Cockroach labs has suggested to disable the leaseholder on the site which is impacted [East or West].

During Firewall change we have observed AMW2-G4 [Amazon West] site was impacted so in below example and commands will be referencing west site.

Steps to disable leaseholder:

Recommendation for Unplanned firewall change - "lightning strike", When not able to connect to 1 of 3 regions where Replication Factor=3

1. Move Traffic away from Impacted Site.
2. Shutdown nodes [Cockroach service] in the region (does not have to be one at a time)
3. When able to steadily connect to the region/firewall issue is resolved:
4. Bring nodes [Cockroach service] back up one at a time, waiting until replicas have quiesced. Monitor Metrics->Replication Dashboard->Replica Quiescence graph
5. Enable back traffic once all looks good on the cluster.

Recommendation for minimum impact on production workload during Planned Firewall Change:

1. Move Traffic away from Site undergoing firewall Maintenance.
2. Disable lease preference for the region so that leases do not stay in region. [Check commands in below table].
3. Stop nodes [Cockroach service] with 15 min of interval.
4. Rebalancing in this way would not affect the cluster as much as when 5 nodes at one time were disconnected from cluster.
5. After firewall change, bring nodes [Cockroach service] back up with 15 min of interval.
6. Re-introduce lease preference and monitor the cluster to ensure all the ranges and leaseholders have migrated back to West
7. Enable traffic back on the Site.

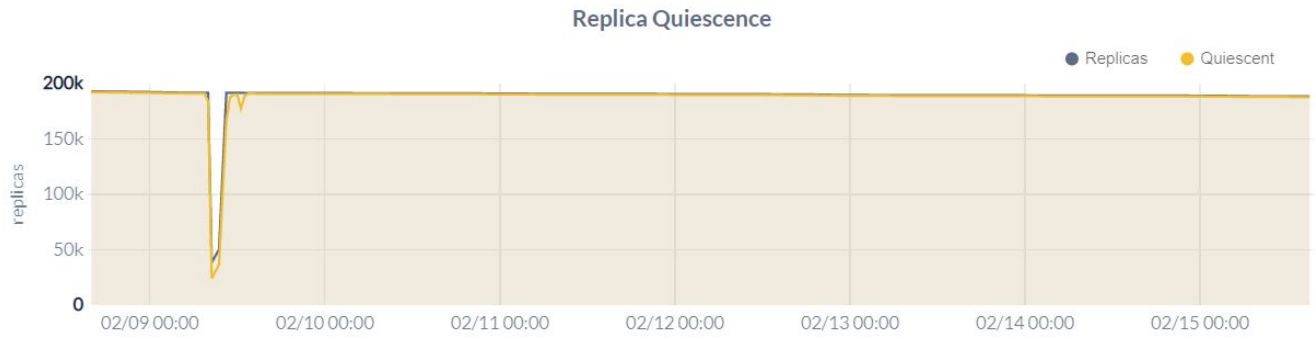
Description	Command
Sample Command to Login to DB. e.g. We are login to Stage Session cluster. We have to use internal name to login as we see in Cluster overview page Prod - ip-10-140-91-117.us-west-2.compute.internal	cd /root/cockroach; cockroach sql --certs-dir=/root/cockroach/certs --host ip-10-140-91-137.us-west-2.compute.internal --port 5432 --database convoy_session_service
Disable Leaseholder Preference. e.g. we are disabling west partition on stage session cluster	ALTER PARTITION west OF INDEX convoy_session_service.public.con_session@primary CONFIGURE ZONE USING lease_preferences = '[]';
To view current partition configuration	SHOW ALL ZONE CONFIGURATIONS;
To Enable Leaseholder preference e.g. We are enabling leaseholder in stage session DB cluster for west partition	ALTER PARTITION west OF INDEX convoy_session_service.public.con_session@primary CONFIGURE ZONE USING lease_preferences = '[[+region=west]]';
This would scatter the leases among the active nodes. **[Run this command only when you don't see lease are settled back to impacted site]**	ALTER TABLE con_session SCATTER;

What we have to monitor during the process ?

Node shutdown

When we shutdown the node [Cockroach service] we have to check the range operation. To check it, open cluster overview page - on left menu select "Metrics" - On Metrics page click on "Dashboard: Overview" and select "Replication" and scroll down to check Range Operation and Snapshots. We need to ensure the range operation in back to normal [zero] before we stop other node.

We have to wait until the replica quiescence is >80% before we stop the next node.



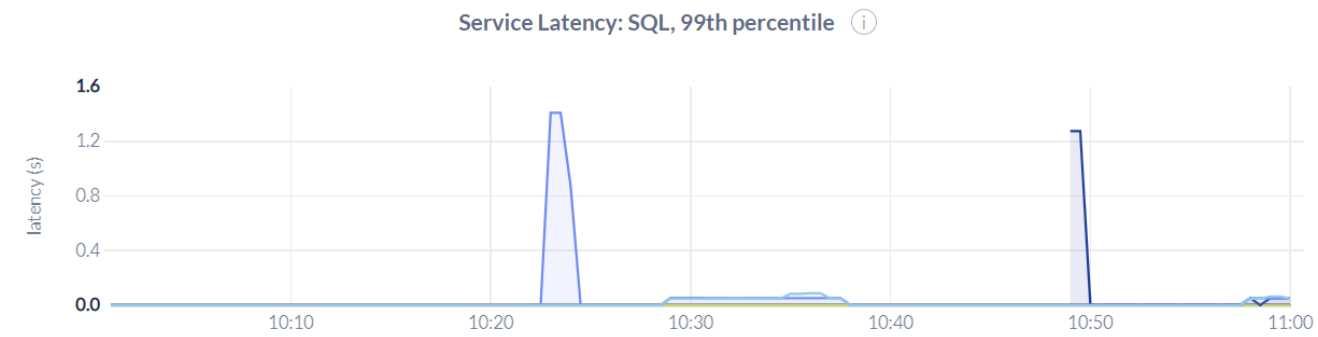
Node Startup

When we start node after enabling "Leaseholder" we have to monitor Range operation, snapshot and SQL connections. To check it, open cluster overview page - on left menu select "Metrics" - On Metrics page click on "Dashboard: Overview" and select "Replication" and scroll down to check Range Operation and Snapshots. We need to ensure the range operation in back to normal [zero] before we start other node.

Bring back all node [Cockroach service] with interval of 15 min and enable back lease holder preference.

We have to wait until the replica quiescence is >80% before we start the next node.

Also ensure you see SQL connection and SQL latency 99 percentile is normal.



Please find below document with list of steps followed with screenshots on what to monitor for metrics/dashboards and command outputs.

It took around 1:20 hrs.' to stop the nodes and 2 hrs.' to start nodes in stage session CRDB cluster

It took around ~20 hrs. to settle range in PROD

Command

Login to cockroach sql

```
[root@ip-10-140-91-137 centos]# cd /root/cockroach; cockroach sql --certs-dir=/root/cockroach/certs --host ip-10-140-91-137.us-west-2.compute.internal --port 5432 --database convoy_session_service
#
# Welcome to the CockroachDB SQL shell.
# All statements must be terminated by a semicolon.
# To exit, type: \q.
#
# Server version: CockroachDB CCL v19.2.9 (x86_64-unknown-linux-gnu, built 2020/06/29 22:02:23, go1.12.12) (same version as client)
# Cluster ID: d1505060-554f-4ae0-ad9c-7ad3b8ce7571
# Organization: Comcast AIQ
#
# Enter \? for a brief introduction.
#
```

Disable west leaseholder

```
root@ip-10-140-91-137.us-west-2.compute.internal:5432/convoy_session_service> ALTER PARTITION west OF INDEX
convoy_session_service.public.con_session@primary CONFIGURE ZONE USING lease_preferences = '[]';

CONFIGURE ZONE 1

Time: 4.198240026s
```

Example - Show configurations after the change

```
root@ip-10-140-91-137.us-west-2.compute.internal:5432/convoy_session_service> SHOW ALL ZONE CONFIGURATIONS;
                                    target
|                                             raw_config_sql
+-------------------------------------------------------------------------
+-------------------------------------------------------------------------------------------------------------+
  RANGE default                                                                 | ALTER RANGE default CONFIGURE
ZONE USING
                                                                                |     range_min_bytes =
1048576,
                                                                                |     range_max_bytes =
67108864,
                                                                                |     gc.ttlseconds =
90000,
                                                                                |     num_replicas =
3,
                                                                                |     constraints =
'[]',
                                                                                |     lease_preferences =
'[]'
  DATABASE system                                                               | ALTER DATABASE system
CONFIGURE ZONE USING
                                                                                |     range_min_bytes =
1048576,
                                                                                |     range_max_bytes =
67108864,
                                                                                |     gc.ttlseconds =
90000,
                                                                                |     num_replicas =
5,
                                                                                |     constraints =
'[]',
                                                                                |     lease_preferences =
'[]'
  TABLE system.public.jobs                                                      | ALTER TABLE system.public.jobs
CONFIGURE ZONE USING
                                                                                |     range_min_bytes =
1048576,
                                                                                |     range_max_bytes =
67108864,
                                                                                |     gc.ttlseconds =
600,
                                                                                |     num_replicas =
5,
                                                                                |     constraints =
'[]',
                                                                                |     lease_preferences =
'[]'
  RANGE meta                                                                    | ALTER RANGE meta CONFIGURE
ZONE USING
                                                                                |     range_min_bytes =
1048576,
                                                                                |     range_max_bytes =
67108864,
                                                                                |     gc.ttlseconds =
3600,
                                                                                |     num_replicas =
```

```
5,
                                                                                |     constraints =
'[]',
                                                                                |     lease_preferences =
'[]'
  RANGE system                                                                  | ALTER RANGE system CONFIGURE
ZONE USING
                                                                                |     range_min_bytes =
1048576,
                                                                                |     range_max_bytes =
67108864,
                                                                                |     gc.ttlseconds =
90000,
                                                                                |     num_replicas =
5,
                                                                                |     constraints =
'[]',
                                                                                |     lease_preferences =
'[]'
  RANGE liveness                                                                | ALTER RANGE liveness CONFIGURE
ZONE USING
                                                                                |     range_min_bytes =
1048576,
                                                                                |     range_max_bytes =
67108864,
                                                                                |     gc.ttlseconds =
600,
                                                                                |     num_replicas =
5,
                                                                                |     constraints =
'[]',
                                                                                |     lease_preferences =
'[]'
  TABLE system.public.replication_constraint_stats                              | ALTER TABLE system.public.
replication_constraint_stats CONFIGURE ZONE USING
                                                                                |     gc.ttlseconds =
600
  TABLE system.public.replication_stats                                         | ALTER TABLE system.public.
replication_stats CONFIGURE ZONE USING
                                                                                |     gc.ttlseconds =
600
  PARTITION east OF INDEX convoy_session_service.public.con_session@primary    | ALTER PARTITION east OF INDEX
convoy_session_service.public.con_session@primary CONFIGURE ZONE USING
                                                                                |     range_min_bytes =
1048576,
                                                                                |     range_max_bytes =
67108864,
                                                                                |     gc.ttlseconds =
90000,
                                                                                |     num_replicas =
3,
                                                                                |     constraints =
'[]',
                                                                                |     lease_preferences =
'[[+region=east]]'
  PARTITION west OF INDEX convoy_session_service.public.con_session@primary    | ALTER PARTITION west OF INDEX
convoy_session_service.public.con_session@primary CONFIGURE ZONE USING
                                                                                |     range_min_bytes =
1048576,
                                                                                |     range_max_bytes =
67108864,
                                                                                |     gc.ttlseconds =
90000,
                                                                                |     num_replicas =
3,
                                                                                |     constraints =
'[]',
                                                                                |     lease_preferences =
'[[+region=west]]'
  PARTITION central OF INDEX convoy_session_service.public.con_session@primary | ALTER PARTITION central OF
INDEX convoy_session_service.public.con_session@primary CONFIGURE ZONE USING
```

```
                                                                    |     range_min_bytes =
1048576,
                                                                    |     range_max_bytes =
67108864,
                                                                    |     gc.ttlseconds =
90000,
                                                                    |     num_replicas =
3,
                                                                    |     constraints =
'[]',
                                                                    |     lease_preferences =
'[[+region=central]]'
```

Example - Alter west lease holder

```
root@ip-10-140-91-137.us-west-2.compute.internal:5432/convoy_session_service> ALTER PARTITION west OF INDEX
convoy_session_service.public.con_session@primary CONFIGURE ZONE USING lease_preferences = '[[+region=west]]';

CONFIGURE ZONE 1

Time: 4.225618391s
```

Kibana Link To Monitor Transient Session Objects Being Created at AIQ Orchestration Service When Create Session Fails - https://vpc-teletraan-prod-dx-w37tka3obiurzokzjc4arwzpmi.us-west-2.es.amazonaws.com/_plugin/kibana/goto/47692f1cbd4dbcf08d32a34a1d1270fb

Comcast Confidential - Any unauthorized disclosure or use is strictly prohibited

Troubleshooting CockroachDB alerts

1. Clock synchronization error on VMs
2. Liveness failed heartbeat or connection error
3. Disk IO latency on VMs

1. Every VM undergoes Vmotioning based on its usage of resources like Memory and CPU cores. During VMotioning, unused resources of VMs are redistributed to other VMs in need. This phenomenon causes drift in Clock time and causes synchronization error, which in turn causes the node to go down on CockroachDB UI. Typically, it takes 20-30 minutes for VMotioning to stabilize. If it was observed multiple times or frequently, we approach VM /Unix team to move the nodes into a different ESX cluster, to avoid frequent VMotioning. Under Hardware section of dropdown in Metrics UI, we can see Clock offset on VMs. **If the offset is beyond 500 milli seconds, thats an indication of Clock sync issue**. If the node is down and not accessible for metrics on UI, you can SSH on to the node and run “ntpq -p” command to look for Offset result. That way we know if it is a Clock sync caused error. Also, if we observe the hostmon based CPU/Memory utilization trends on VMs, you can understand if the node has undergone VMotion. With a window of drastic decrease of VM memory, it is an indication that VM ESX has repurposed this node resources to some other VM and usually stabilizes within few minutes.

```
ntpq -p
```

2. During a window of 1AM to 6 AM, all Firewall related changes are implemented. This can cause network blips on VM nodes. There is no alternative solution for Session Service or CMS VM nodes. If it undergoes a network blip, the usual turn-around time is within 5 - 10 minutes. In other words, an alert indicating a “liveness node count” decrease will auto-recover and a follow-up Healthy notification arrives within 10 minutes. This, if occurred beyond the regular window, we need to dig through the logs on any of the node to further understand underlying issue. Command to check for these errors:

```
egrep 'failed node heartbeat | connection error' /root/cockroach/cockroach-data/logs/cockroach.log
```

3. Like in the case of Vmotioning, we shall see I/O commit latency drift on VM nodes. This can be viewed on Cockroachdb metrics under SQL menu. Disk reads and writes on VM shall trigger these kind of issues.

Troubleshooting Procedures - CRDB Cluster Response Time Spikes Causing AIQ Service Timeouts

Impact :

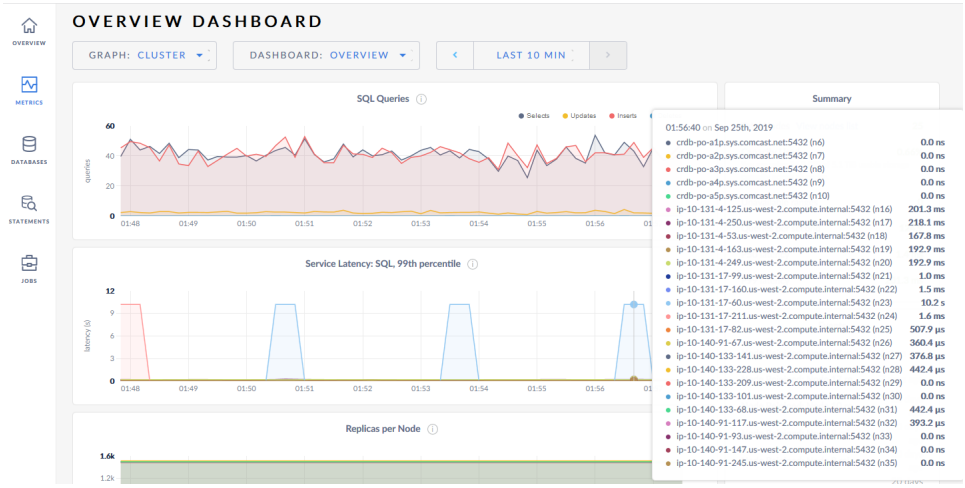
aIQ services can experience timeouts

Possible causes

- The continuous occurrence of p99 latency, p99 latency can occur due to one or more SQL queries taking more time than the usual for execution.
- High Disk IOPS being registered due to handle raft ready messages logged on the CRDB nodes.
- Increase in log commit latency
- Spikes in CPU utilization.

Troubleshooting steps :

- Identify the timeouts Experiencing on the aIQ services are caused due to latency in session service or user service CRDB nodes.
- Once identified, navigate to the metrics by logging in to the respective session service or user service CRDB admin URL.
  session service Admin URL: https://internal-aiqcrdb-ss-prd-elb-823876589.us-east-1.elb.amazonaws.com:8080/#/overview/list login: crdbui ; password : <Refer AIQ Vault>
  user service Admin URL: https://internal-aiqcrdb-ss-prd-elb-823876589.us-east-1.elb.amazonaws.com:8080/#/overview/list login : crdbui ; password : <Refer AIQ Vault>
- check the metrics for Service Latency: SQL, 99th percentile in the CRDB admin console and identify any CRDB nodes causing high latency within the respective time frame.



- Similarly, check the metrics for Disk IOPS In Progress in CRDB Admin console under hardware filter and check for any abnormal pattern in the disk IOPS.

- check the metrics for Log Commit Latency: 99th Percentile with the filter as storage, ideally the log commit latency should be in a single-digit ms, identify for any CRDB nodes exhibiting high log commit latency.
- Identify the respective CRDB nodes from the above scenarios and login into the respective nodes and check the logs check for any abnormal behavior in logs. if logs on the respective CRDB nodes look normal, obtain the debug.zip and share with the cockroach labs to troubleshoot further.

Post troubleshooting validation :

- check the metrics for Service Latency: SQL, 99th percentile, Disk IOPS, Log Commit Latency: 99th Percentile are back to normal.

Communication :

- Acknowledge the issue initially and post updates in **#aiq-convoy slack channel.**
- Post updates in **#aiq-convoy** every 30 min regarding any background of the issue and troubleshooting steps.
- Post final round of updates with RCA or summary regarding the issue in **#aiq-convoy slack** once the issue is resolved.

Troubleshooting Procedures - PCF To AIQ Backend Endpoints Connectivity Issues

Impact :

aIQ services will experience timeouts due to connectivity issues with backend endpoints.

Possible causes :

1. Experiencing Connectivity issues from PCF env ame1-g3 and amw2-g3 to new respective backend endpoint aIQ services are consuming.
2. Experiencing intermittent Connectivity issues from PCF env ame1-g3 and amw2-g3 to existing backend endpoint aIQ services are consuming.

Troubleshooting steps :

1. **Experiencing Connectivity issues from PCF env ame1-g3 and amw2-g3 to new respective backend endpoint aIQ services are consuming.**

- Initially identify the source and destination end-points, the source would be the PCF CIDR block IP range on AWS 10.131.0.0/23 and destination can be respective backend endpoint aIQ services are consuming.
- Validate connectivity from P2 | G7 | AME1-G3 | AMW2-G3 PCF sites to respective endpoint needed to establish connectivity.
  Run the below URL's in the browser for testing connectivity from respective PCF sites on ports 80 and 443.

PCF Env	URL for testing connectivity
P2	http://isconnected.p2.app.cloud.comcast.net/v1/$host/443 http://isconnected.p2.app.cloud.comcast.net/v1/$host/80
G7	http://isconnected.g7.app.cloud.comcast.net/v1/$host/443 http://isconnected.g7.app.cloud.comcast.net/v1/$host/80
AME1-G3	http://isconnected.ame1-g3.cf.comcast.net/v1/$host/443 http://isconnected.ame1-g3.cf.comcast.net/v1/$host/80
AMW2-G3	http://isconnected.amw2-g3.cf.comcast.net/v1/$host/443 http://isconnected.amw2-g3.cf.comcast.net/v1/$host/80

Note: Replace $host in the above URL with respective Endpoints/IP address needed to establish connectivity.

Connection successful scenario

for instance, let's validate connectivity to endpoint tts-service.g.comcast.net from P2 | G7 | AME1-G3 | AMW2-G3 PCF sites

http://isconnected.p2.app.cloud.comcast.net/v1/tts-service.g.comcast.net/443

```
{"status":true,"message":"Connection to tts-service.g.comcast.net port 443 succeeded!"}
```

http://isconnected.g7.app.cloud.comcast.net/v1/tts-service.g.comcast.net/443

```
{"status":true,"message":"Connection to tts-service.g.comcast.net port 443 succeeded!"}
```

http://isconnected.g7.app.cloud.comcast.net/v1/tts-service.g.comcast.net/443

```
{"status":true,"message":"Connection to tts-service.g.comcast.net port 443 succeeded!"}
```

http://isconnected.g7.app.cloud.comcast.net/v1/tts-service.g.comcast.net/443

```
{"status":true,"message":"Connection to tts-service.g.comcast.net port 443 succeeded!"}
```

Connection failure scenario

Considering below use case for establishing connectivity to the endpoint esp-int.cable.comcast.com from G7 PCF env.
step1: validate connectivity from G7 PCF env using http://isconnected.g7.app.cloud.comcast.net/v1/esp-int.cable.comcast.com/443

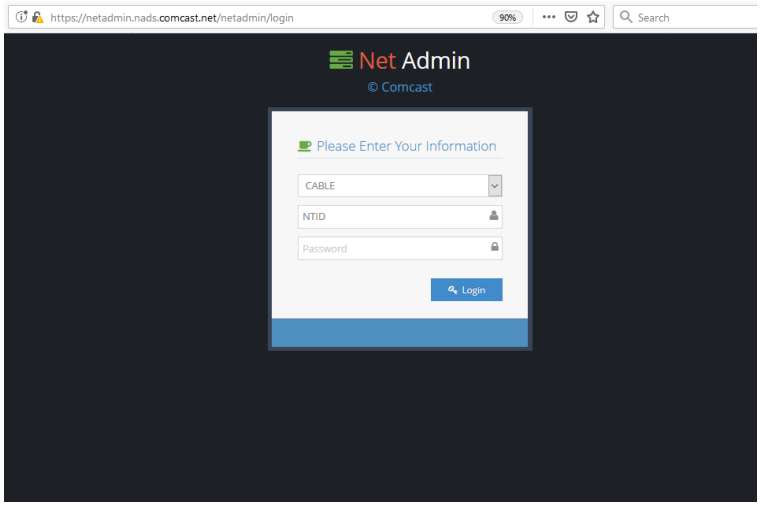
{"status":false,"message":"*** Connection timed out - connect(2) for esp-int.cable.comcast.com port 443"}

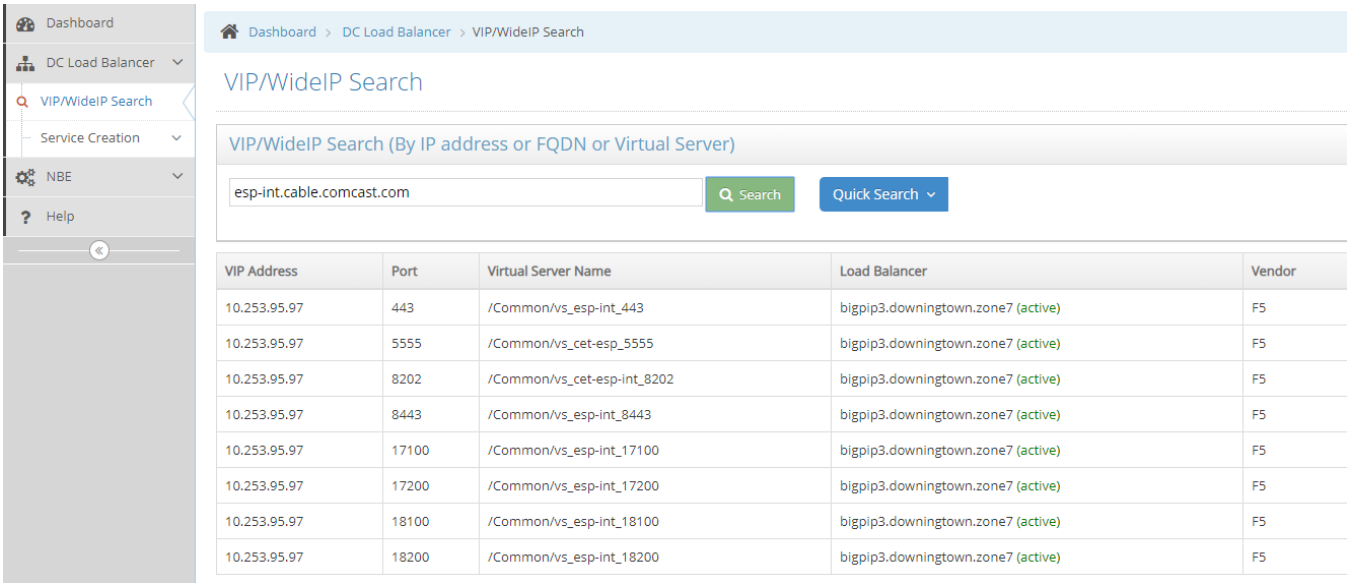
step2: Identify the source and destination IP ranges for implementing an SC Firewall ticket.

step 3: Refer to the table at the end section for CIDR ranges for PCF subnets, source: 10.144.77.0/24 (G7)

To identify the IP address ranges for destination endpoint esp-int.cable.comcast.com refer to the netadmin portal, NetAdmin is a user interface platform that hosts various applications and can get endpoint details of different applications.

step 4: access NetAdmin using the following URL. **https://netadmin.nads.comcast.net** and navigate to Dashboard < DC Load Balancer < VIP /WideIP Search and lookup for esp-int.cable.comcast.com

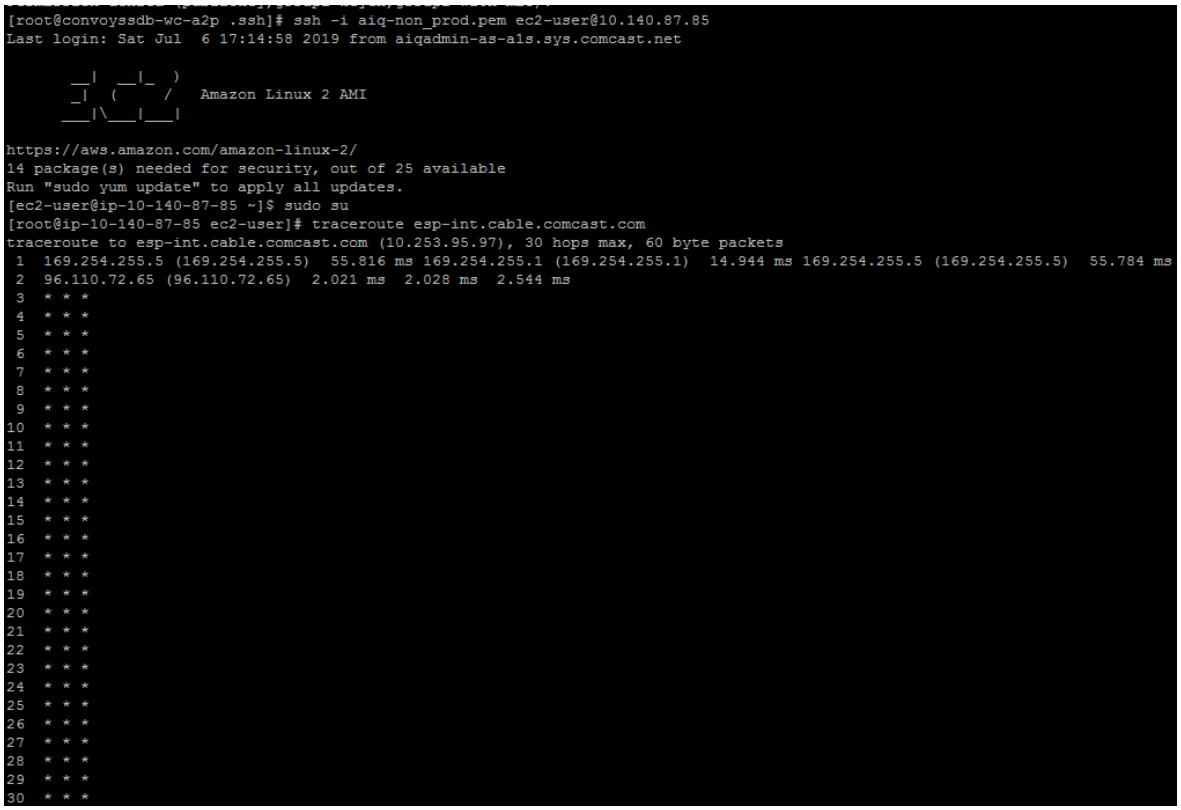




step 6: Obtain the destination IP address for the respective Endpoint esp-int.cable.comcast.com from the netadmin portal as shown above and Request an FW SC Ticket https://servicecatalog.cable.comcast.com/sc/catalog.product.aspx?product_id=fw_acl_add_change with the source and destination endpoints on ports 80 and 443, along with any other exclusive ports needed to be included in the FW ticket.

step 7: Validate connectivity using step1 once the FW ticket had been implemented and if the connectivity still fails to open a bridge with the TPX Firewall Ops TPX-Firewall_Ops@comcast.com and share the traceroute and telnet results for the respective destination end-point.

step 8: log in to the ec2 instance DX-TEST-86 | 10.140.87.85 in AME1-G3 (DX-INT-EAST) under PCF subnet range and perform a traceroute and telnet on the destination Endpoint.



2. Experiencing intermittent Connectivity issues from PCF env ame1-g3 and amw2-g3 to existing backend endpoint aIQ services are consuming.

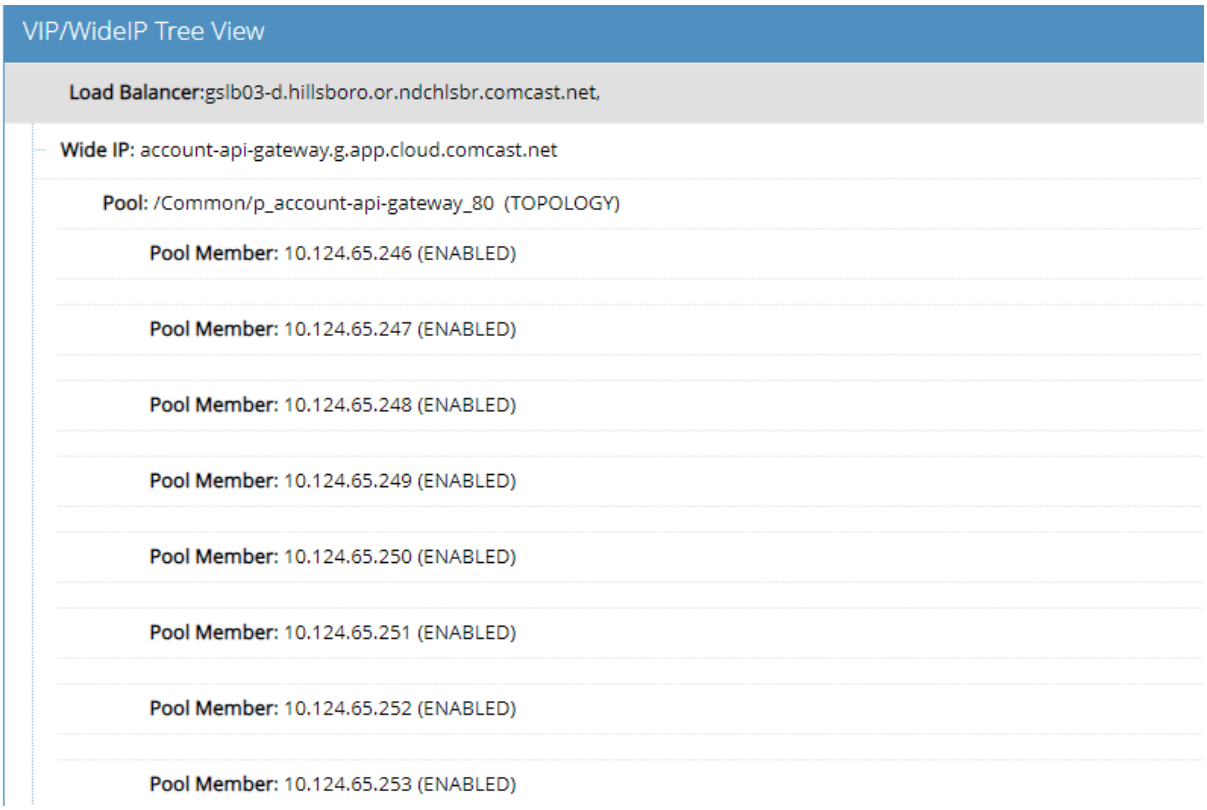
step 1: Initially identify the source and destination end-points, the source would be the respective PCF CIDR block IP range on AWS 10.140.86.0/23 (AME1-G3, east) and 10.140.137.0/24 (AMW2-G3, west) and destination can be respective backend endpoint aIQ services are consuming.

step 2: Validate connectivity from AME1-G3 | AMW2-G3 PCF sites to respective endpoint needed to establish connectivity, Run the below URLs in the browser for testing connectivity from respective east and west PCF sites on ports 80 and 443 and verify if connectivity to the respective backend endpoints failing intermittently.

PCF Env	URL for testing connectivity
AME1-G3	http://isconnected.ame1-g3.cf.comcast.net/v1/$host/443 http://isconnected.ame1-g3.cf.comcast.net/v1/$host/80
AMW2-G3	http://isconnected.amw2-g3.cf.comcast.net/v1/$host/443 http://isconnected.amw2-g3.cf.comcast.net/v1/$host/80

step 3: Identify the destination end-point and list out the IP address of the pool members under the respective Wide IP , let us consider the destination end-point as account-api-gateway.g.app.cloud.comcast.net

step 4: List out the IP address for account-api-gateway.g.app.cloud.comcast.net using Netadmin portal



step 4: Login to the ec2 instance **10.131.17.45** in us-west and navigate to **/root/cockroach/scripts** and update the hosts-curl file with the respective IP address for the destination endpoint and run the respective scripts to test connectivity from the east (curl-east-ame1.sh) and west

(curl-west-amw2.sh) and validate connectivity on port 80 and 443 .

```
[root@ip-10-131-17-45 scripts]# ./curl-east-ame1.sh
{"status":true,"message":"Connection to 10.124.65.246 port 443 succeeded!"}{"status":true,"message":"Connection to 10.124.65.247 port 443
ue,"message":"Connection to 10.124.65.249 port 443 succeeded!"}{"status":true,"message":"Connection to 10.124.65.250 port 443 succeeded!"}
```

step 5: check for any missing connectivity and repeat steps 6 - 8 as necessary.

Post troubleshooting validation :

Validate connectivity from AME1-G3 | AMW2-G3 PCF sites to respective endpoint needed to establish connectivity, Run the below URLs in the browser for testing connectivity from respective east and west PCF sites on ports 80 and 443 and verify if connectivity Exists.

PCF Env	URL for testing connectivity
AME1-G3	http://isconnected.ame1-g3.cf.comcast.net/v1/$host/443 http://isconnected.ame1-g3.cf.comcast.net/v1/$host/80
AMW2-G3	http://isconnected.amw2-g3.cf.comcast.net/v1/$host/443 http://isconnected.amw2-g3.cf.comcast.net/v1/$host/80

Communication:

- Acknowledge the issue initially and post updates in **#aiq-va-ops slack channel.**
- Post updates in **#aiq-va-ops** every 30 min regarding any background of the issue and troubleshooting steps.
- Post final round of updates with RCA or summary regarding the issue in **#aiq-va-ops slack** once the issue is resolved.

PCF subnets info P2 | G7 | AME1-G3 | AMW2-G3 :

PCF Env	CIDR Range
P2	10.131.0.0/23
G7	10.144.77.0/24
AME1-G3	10.140.86.0/23
AMW2-G3	10.140.137.0/24

Troubleshooting Procedures - Repeated Invalid 1111111111111111 Account Number Used to Call for Appointments

Symptom

Repeated calls to csp ***/einstein/rest/selfhelp/account/1111111111111111/appointment*** with client ID ***aiq_cx_tools***. There may be other client IDs too.

With the repeated call to appointments, hitting CSG, CSG apps will have issues, especially when there's 6k worth of orders on that account.

Cause

In Magic Eraser flow, whenever an event comes into WFA's Kafka topic "workorder-topic", a call will be fired to csp see details of the appointment on the account. There is no retry.

https://streamva-po-e01p.sys.comcast.net:5601/goto/2956476b9fc27eea047ea2e5f70b0122

In the case where invalid account numbers are in the event, invalid account number will be used to call csp.

In the incident on May. 20th, 2021, a biller macro "D+G M1H2 NED MACRO ID" is writing a lot of workorders to the invalid account, causing huge amount of events writing to that kafka topic.

In #wfa-apps-support Slack channel, Ben Henry and Matthew Maurer helped reached out to the macro owners.

Method to Mitigate

The NiFi flow that calls CSP after getting the event from WFA topic is on streamcxin NiFi server. Girish Nair is the contact. If we see repeated invalid account calls, then we need to filter out the account from calling csp.

Comcast Confidential - Any unauthorized disclosure or use is strictly prohibited