Restore Medusa Backup - IN-Place

Apache medusa gives us an ability to perform restore of data on the same node where the backup was taken termed as **"in-place"** restore and second type of restore where data can be restored on a different node than on which the backup was taken.

This article deals with the in-place backup restore. Below are the steps for the same.

This article will be covered with a real-time example for better understanding.

Step 1 : Check the status and health of the cluster before beginning the activity.

```
root@poojatestcassandra1-az-prod-ci:~# nodetool status
root@poojatestcassandra1-az-prod-ci:~# nodetool status
Datacenter: ppsCI
=================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address       Load        Tokens  Owns  Host ID                               Rack
UN  10.12.74.152  1.73 GiB    256     ?     74bcda0a-8adf-4d69-b9d6-f46d88471a70  rack1
UN  10.12.72.148  2.14 GiB    256     ?     1c0d75da-c137-4644-9b89-51655856bf76  rack1
UN  10.12.74.241  897.33 MiB  256     ?     be670902-f45a-4826-b687-bcd01614b126  rack1
UN  10.12.72.141  2.96 GiB    256     ?     03462c50-7f9e-4aa3-af8a-dac71f2f1769  rack1
```

Step 2: Check the data present in the cluster

```
cassandra@cqlsh> desc KEYSPACES ;

keyspace1  system       system_distributed  system_traces  system_virtual_schema
pps        system_auth  system_schema       system_views
```

```
cassandra@cqlsh> SELECT * FROM pps.deferred_plan_data  ;

 ppsid | cartresponse | comments | createdtimestamp | orderid | paynowrequest | retrycounter | tenantid |
updatedby | updatedtimestamp | xid
-------+--------------+----------+------------------+---------+---------------+--------------+----------
+-----------+------------------+-----

(0 rows)
```

Step 3 : Insert some data to reflect the restore once done.

```
cassandra@cqlsh:pps> INSERT INTO deferred_plan_data (ppsid , cartresponse , comments , createdtimestamp ,
orderid , paynowrequest , retrycounter , tenantid , updatedby , updatedtimestamp , xid ) VALUES ( '1234' ,
'Good' , 'Good work' , toTimeStamp(toDate(now())) , '1', 'test' , 1 , '1234' , 'Pooja' , toTimeStamp(toDate
(now())) , '1234' ) ;
cassandra@cqlsh:pps> SELECT * FROM pps.deferred_plan_data  ;


cassandra@cqlsh:pps> INSERT INTO deferred_plan_data (ppsid , cartresponse , comments , createdtimestamp ,
orderid , paynowrequest , retrycounter , tenantid , updatedby , updatedtimestamp , xid ) VALUES ( '5678' ,
'Good' , 'Bad work' , toTimeStamp(toDate(now())) , '2', 'test-2' , 2 , '5678' , 'Pooja N' , toTimeStamp(toDate
(now())) , '5678' ) ;
```

```
cassandra@cqlsh:pps> SELECT * FROM pps.deferred_plan_data ;
 ppsid | cartresponse | comments  | createdtimestamp | orderid | paynowrequest | retrycounter | tenantid |
updatedby | updatedtimestamp | xid
-------+--------------+-----------+------------------+---------+---------------+--------------+----------
+-----------+------------------+------
  5678 |         Good |  Bad work |    1747699200000 |       2 |        test-2 |            2 |     5678 |
Pooja N |    1747699200000 | 5678
  1234 |         Good | Good work |    1747699200000 |       1 |          test |            1 |     1234 |
Pooja |    1747699200000 | 1234

(2 rows)
```

Step 4: Restore the older data to the cluster whereis not present.

> **IMPORTANT** : While running restore, make sure to give a seed node IP so that other nodes wait for this seed to boot first and rest nodes to follow - This a recommendation provided by the community for a clean restore. Also to run the restore on all the nodes at the same time.

Run the below restore command for an in-place restore. (Include seed node info for all the nodes and exclude the seed info for the one seed node mentioned for other nodes.)

Example - The seed node of this cluster is - **10.12.72.148**

Restore command for all the other nodes would be as below

medusa --prefix PPS-test-cluster-cassandra_WEEK_21_2025 restore-node --in-place --backup-name backup-20-May-2025-13-50 --seeds 10.12.72.148

Restore command for seed node - **10.12.72.148** would be as below

medusa --prefix PPS-test-cluster-cassandra_WEEK_21_2025 restore-node --in-place --backup-name backup-20-May-2025-13-50

Logs for the Restore

Start of restore :

```
[2025-05-20 14:30:01,858] DEBUG: Loading configuration from /etc/medusa/medusa.ini
[2025-05-20 14:30:01,862] INFO: Resolving ip address
[2025-05-20 14:30:01,863] INFO: ip address to resolve 10.12.74.241
[2025-05-20 14:30:01,863] DEBUG: Not resolving 10.12.74.241 as requested
[2025-05-20 14:30:01,864] DEBUG: Logging to file options: LoggingConfig(enabled='1', file='/var/log/medusa
/medusa.log', format='[%(asctime)s] %(levelname)s: %(message)s', level='INFO', maxBytes='20000000',
backupCount='50')
[2025-05-20 14:30:01,864] DEBUG: Loading storage_provider: azure_blobs
[2025-05-20 14:30:01,864] DEBUG: Loading identity from credentials file: /etc/medusa/medusa-azure-credentials
[2025-05-20 14:30:01,865] DEBUG: [Storage] Getting object 10.12.74.241/backup-20-May-2025-13-50/meta
/differential
[2025-05-20 14:30:01,865] DEBUG: Using selector: GeventSelector
[2025-05-20 14:30:01,959] DEBUG: Blob PPS-test-cluster-cassandra_WEEK_21_2025/10.12.74.241/backup-20-May-2025-
13-50/meta/schema.cql was not found in cache.
[2025-05-20 14:30:01,959] DEBUG: [Storage] Getting object PPS-test-cluster-cassandra_WEEK_21_2025/10.12.74.241
/backup-20-May-2025-13-50/meta/schema.cql
[2025-05-20 14:30:01,998] DEBUG: [Storage] Getting object PPS-test-cluster-cassandra_WEEK_21_2025/10.12.74.241
/backup-20-May-2025-13-50/meta/manifest.json
[2025-05-20 14:30:02,037] DEBUG: [Storage] Reading blob PPS-test-cluster-cassandra_WEEK_21_2025/10.12.74.241
/backup-20-May-2025-13-50/meta/manifest.json...
[2025-05-20 14:30:02,126] DEBUG: This server has systemd: True
[2025-05-20 14:30:02,271] INFO: Downloading data from backup to /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-
7f512251cee9
[2025-05-20 14:30:02,272] DEBUG: Download size: 812607566, available space: 54064508928
[2025-05-20 14:30:02,272] DEBUG: Loading storage_provider: azure_blobs
[2025-05-20 14:30:02,273] DEBUG: Loading identity from credentials file: /etc/medusa/medusa-azure-credentials
```

Download the backed up data into a temp location

```
[2025-05-20 14:30:02,275] DEBUG: Downloading  8 files to /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-
7f512251cee9/system_auth/resource_role_permissons_index-5f2fbdad91f13946bd25d5da3a5c35ec
[2025-05-20 14:30:02,373] DEBUG: [Azure Storage] Downloading with 1 workers: medusa-test-bucket/PPS-test-
cluster-cassandra_WEEK_21_2025/10.12.74.241/data/system_auth/resource_role_permissons_index-
5f2fbdad91f13946bd25d5da3a5c35ec/nb-1-big-Summary.db -> /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-7f512251cee9
/system_auth/resource_role_permissons_index-5f2fbdad91f13946bd25d5da3a5c35ec/nb-1-big-Summary.db
[2025-05-20 14:30:02,471] DEBUG: [Azure Storage] Downloading with 1 workers: medusa-test-bucket/PPS-test-
cluster-cassandra_WEEK_21_2025/10.12.74.241/data/system_auth/resource_role_permissons_index-
5f2fbdad91f13946bd25d5da3a5c35ec/nb-1-big-Index.db -> /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-7f512251cee9
/system_auth/resource_role_permissons_index-5f2fbdad91f13946bd25d5da3a5c35ec/nb-1-big-Index.db
[2025-05-20 14:30:02,547] DEBUG: [Azure Storage] Downloading with 1 workers: medusa-test-bucket/PPS-test-
cluster-cassandra_WEEK_21_2025/10.12.74.241/data/system_auth/resource_role_permissons_index-
5f2fbdad91f13946bd25d5da3a5c35ec/nb-1-big-TOC.txt -> /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-7f512251cee9
/system_auth/resource_role_permissons_index-5f2fbdad91f13946bd25d5da3a5c35ec/nb-1-big-TOC.txt
[2025-05-20 14:30:02,605] DEBUG: [Azure Storage] Downloading with 1 workers: medusa-test-bucket/PPS-test-
cluster-cassandra_WEEK_21_2025/10.12.74.241/data/system_auth/resource_role_permissons_index-
5f2fbdad91f13946bd25d5da3a5c35ec/nb-1-big-Digest.crc32 -> /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-
7f512251cee9/system_auth/resource_role_permissons_index-5f2fbdad91f13946bd25d5da3a5c35ec/nb-1-big-Digest.crc32
```

Stop the Cassandra process and wait for it to be completely down

```
[2025-05-20 14:31:03,314] DEBUG: Disconnecting from Azure Storage
[2025-05-20 14:31:03,315] INFO: Stopping Cassandra
[2025-05-20 14:31:13,025] INFO: Waiting for Cassandra to go down on 10.12.74.241
[2025-05-20 14:31:13,026] INFO: Verifying node state for host 10.12.74.241 using check type cql
[2025-05-20 14:31:13,036] DEBUG: This server has systemd: True
[2025-05-20 14:31:13,136] DEBUG: Checking Cassandra health type: cql for host: 10.12.74.241 release_ver: 3.11.9
native_port: 9042 storage_port: 7000, rpc_port: 9160
[2025-05-20 14:31:13,137] DEBUG: Port '9042' is closed, assuming '10.12.74.241' is down.
[2025-05-20 14:31:13,137] DEBUG: The node 10.12.74.241 is not up yet...
[2025-05-20 14:31:13,137] DEBUG: Port '7000' is closed, assuming '10.12.74.241' is down.
[2025-05-20 14:31:13,137] DEBUG: The node 10.12.74.241 is not up yet...
[2025-05-20 14:31:13,137] INFO: Cassandra is down 10.12.74.241
```

Cleanup the present data and move the backedup data to the original location

```
[2025-05-20 14:31:13,137] DEBUG: Cleaning (/myntra/cassandra/commitlog)
[2025-05-20 14:31:13,137] DEBUG: Removing files - keep folder /myntra/cassandra/commitlog
[2025-05-20 14:31:13,137] DEBUG: Removing file /myntra/cassandra/commitlog/CommitLog-7-1747727267604.log
[2025-05-20 14:31:13,154] DEBUG: Blob PPS-test-cluster-cassandra_WEEK_21_2025/10.12.74.241/backup-20-May-2025-
13-50/meta/server_version.json was not found in cache.
[2025-05-20 14:31:13,154] DEBUG: [Storage] Getting object PPS-test-cluster-cassandra_WEEK_21_2025/10.12.74.241
/backup-20-May-2025-13-50/meta/server_version.json
[2025-05-20 14:31:13,274] DEBUG: [Storage] Reading blob PPS-test-cluster-cassandra_WEEK_21_2025/10.12.74.241
/backup-20-May-2025-13-50/meta/server_version.json...
[2025-05-20 14:31:13,302] INFO: Moving backup data to Cassandra data directory
[2025-05-20 14:31:13,302] DEBUG: Cleaning directory /myntra/cassandra/data/system_auth/network_permissions-
d46780c22f1c3db9b4c1b8d9fbc0cc23
[2025-05-20 14:31:13,318] DEBUG: Skipping the actual restore of network_permissions-
d46780c22f1c3db9b4c1b8d9fbc0cc23 - table empty
[2025-05-20 14:31:13,318] DEBUG: Cleaning directory /myntra/cassandra/data/system_auth
/resource_role_permissons_index-5f2fbdad91f13946bd25d5da3a5c35ec
[2025-05-20 14:31:13,332] DEBUG: Restoring /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-7f512251cee9/system_auth
/resource_role_permissons_index-5f2fbdad91f13946bd25d5da3a5c35ec -> /myntra/cassandra/data/system_auth
/resource_role_permissons_index-5f2fbdad91f13946bd25d5da3a5c35ec
[2025-05-20 14:31:13,360] DEBUG: Cleaning directory /myntra/cassandra/data/system_auth/role_members-
0ecdaa87f8fb3e6088d174fb36fe5c0d
[2025-05-20 14:31:13,377] DEBUG: Skipping the actual restore of role_members-0ecdaa87f8fb3e6088d174fb36fe5c0d -
table empty
[2025-05-20 14:31:13,377] DEBUG: Cleaning directory /myntra/cassandra/data/system_auth/role_permissions-
3afbe79f219431a7add7f5ab90d8ec9c
[2025-05-20 14:31:13,391] DEBUG: Restoring /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-7f512251cee9/system_auth
/role_permissions-3afbe79f219431a7add7f5ab90d8ec9c -> /myntra/cassandra/data/system_auth/role_permissions-
3afbe79f219431a7add7f5ab90d8ec9c


[2025-05-20 14:31:13,901] DEBUG: Cleaning directory /myntra/cassandra/data/pps/refund_data-
b98a96a0c56f11e9a4b5312d12ffaad3
[2025-05-20 14:31:14,043] DEBUG: Restoring /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-7f512251cee9/pps
/refund_data-b98a96a0c56f11e9a4b5312d12ffaad3 -> /myntra/cassandra/data/pps/refund_data-
b98a96a0c56f11e9a4b5312d12ffaad3
[2025-05-20 14:31:14,704] DEBUG: Cleaning directory /myntra/cassandra/data/pps/deferred_sync_plan_data-
97427ef0c56f11e9a4b5312d12ffaad3
[2025-05-20 14:31:14,742] DEBUG: Restoring /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-7f512251cee9/pps
/deferred_sync_plan_data-97427ef0c56f11e9a4b5312d12ffaad3 -> /myntra/cassandra/data/pps/deferred_sync_plan_data-
97427ef0c56f11e9a4b5312d12ffaad3
[2025-05-20 14:31:14,886] DEBUG: Cleaning directory /myntra/cassandra/data/pps/deferred_plan_data-
739e59b0c56f11e9a4b5312d12ffaad3
[2025-05-20 14:31:14,906] DEBUG: Restoring /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-7f512251cee9/pps
/deferred_plan_data-739e59b0c56f11e9a4b5312d12ffaad3 -> /myntra/cassandra/data/pps/deferred_plan_data-
739e59b0c56f11e9a4b5312d12ffaad3
```

Verify if the seed node is up or else wait for the seed node to be available

```
[2025-05-20 14:31:15,579] INFO: Verifying node state for host 10.12.72.148 using check type cql
[2025-05-20 14:31:15,588] DEBUG: This server has systemd: True
[2025-05-20 14:31:15,693] DEBUG: Checking Cassandra health type: cql for host: 10.12.72.148 release_ver: 3.11.9
native_port: 9042 storage_port: 7000, rpc_port: 9160
[2025-05-20 14:31:15,694] INFO: At least one seed is now up
```

Start the cassandra process and cleanup the temp location

```
[2025-05-20 14:31:15,694] DEBUG: Starting Cassandra with ['/etc/init.d/cassandra', 'start']
[2025-05-20 14:31:19,703] DEBUG: [Storage] Reading blob PPS-test-cluster-cassandra_WEEK_21_2025/10.12.74.241
/backup-20-May-2025-13-50/meta/server_version.json...
[2025-05-20 14:31:19,726] DEBUG: Cleaning (/tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-7f512251cee9)
[2025-05-20 14:31:19,726] DEBUG: Remove folder /tmp/medusa-restore-db1ce2d6-16c0-4647-8a28-7f512251cee9 and
content
[2025-05-20 14:31:19,743] DEBUG: Disconnecting from Azure Storage
```

Step 5 : Validate the cluster

```
root@poojatestcassandra1-az-prod-ci:~# nodetool status
Datacenter: ppsCI
=================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address       Load       Tokens  Owns (effective)  Host ID                               Rack
UN  10.12.74.152  1.59 GiB   256     74.4%             337462b4-86b0-4fe0-976c-ad257be4d825  rack1
UN  10.12.72.148  1.71 GiB   256     69.6%             d0a74f05-34bf-408f-966f-ce3d72738f83  rack1
UN  10.12.74.241  775.3 MiB  256     72.5%             156c3db0-65a6-42ee-b8a7-afa07e72eebe  rack1
UN  10.12.72.141  2.94 GiB   256     83.5%             5bca36e5-67d7-4784-8720-a2f9d2fc0a83  rack1
```

Validate if the newly inserted data is present in the older backup

```
cassandra@cqlsh> use pps ;
cassandra@cqlsh:pps> SELECT * FROM pps.deferred_plan_data ;

 ppsid | cartresponse | comments | createdtimestamp | orderid | paynowrequest | retrycounter | tenantid |
updatedby | updatedtimestamp | xid
-------+--------------+----------+------------------+---------+---------------+--------------+----------
+-----------+------------------+-----

(0 rows)
```