Transferring data between volumes

From time to time, you may want to transfer data from one persistent storage volume to another, while keeping your data within the same Cleura Cloud region.

For example, you might prefer to select a volume type that has become newly available in that region, but find the downtime associated with retyping a single volume prohibitive. In this case, you can choose an on-line synchronization approach, which comes with much reduced downtime.

Prerequisites

Creating volumes from snapshots (and optionally retyping them) requires using the OpenStack CLI, so make sure you have it enabled.

Checking the source volume's state

Assume you have a volume named `sourcevol` that is currently attached to a server named `testsrv`:

```
$ openstack volume list --long
+--------------+-----------+--------+------+---------+----------+--------------+------------+
| ID           | Name      | Status | Size | Type    | Bootable | Attached to  | Properties |
+--------------+-----------+--------+------+---------+----------+--------------+------------+
| 526f7741-2c44 | sourcevol | in-use |   50 | default | false    | Attached to  |            |
| -4d04-a0c8-86 |           |        |      |         |          | testsrv on   |            |
| b8d039e674    |           |        |      |         |          | /dev/vdb     |            |
+--------------+-----------+--------+------+---------+----------+--------------+------------+
```

In this example, the volume status is `in-use` (meaning the volume is currently attached to a server), and the volume type is `default`.

Taking a snapshot of the source volume

First take a *snapshot* (a consistent, read-only, point-in-time copy) of your source volume. Note that since you are taking a snapshot of an `in-use` volume, you need to use the `--force` option with the following command:

```
$ openstack volume snapshot create --force --volume sourcevol sourcevol-snap
+-------------+--------------------------------------+
| Field       | Value                                |
+-------------+--------------------------------------+
```

```
| created_at  | 2023-01-05T14:34:46.409705           |
| description | None                                 |
| id          | 05ece580-4fb0-45dd-96e8-06a46399c38e |
| name        | sourcevol-snap                       |
| properties  |                                      |
| size        | 50                                   |
| status      | creating                             |
| updated_at  | None                                 |
| volume_id   | 526f7741-2c44-4d04-a0c8-86b8d039e674 |
+-------------+--------------------------------------+
```

The snapshot status should change from `creating` to `available` in a matter of seconds. You can subsequently read back its state with the following command:

```
$ openstack volume snapshot show sourcevol-snap
+-------------------------------------------+--------------------------------------+
| Field                                     | Value                                |
+-------------------------------------------+--------------------------------------+
| created_at                                | 2023-01-05T14:34:46.000000           |
| description                               | None                                 |
| id                                        | 05ece580-4fb0-45dd-96e8-06a46399c38e |
| name                                      | sourcevol-snap                       |
| properties                                |                                      |
| size                                      | 50                                   |
| status                                    | available                            |
| updated_at                                | 2023-01-05T14:34:49.000000           |
| volume_id                                 | 526f7741-2c44-4d04-a0c8-86b8d039e674 |
+-------------------------------------------+--------------------------------------+
```

Creating the target volume

Once you have created a snapshot of your source volume, you can use it to preseed your target volume:

```
$ openstack volume create --snapshot sourcevol-snap targetvol
+--------------------+--------------------------------------+
| Field              | Value                                |
+--------------------+--------------------------------------+
| attachments        | []                                   |
| availability_zone  | nova                                 |
| bootable           | false                                |
| consistencygroup_id | None                                |
| created_at         | 2023-01-05T14:44:10.132096           |
| description        | None                                 |
| encrypted          | False                                |
| id                 | c8e0ff43-f837-478b-ad42-1bdb20dbbdb3 |
| multiattach        | False                                |
| name               | targetvol                            |
| properties         |                                      |
| replication_status | None                                 |
| size               | 50                                   |
| snapshot_id        | 05ece580-4fb0-45dd-96e8-06a46399c38e |
```

```
| source_volid        | None                                 |
| status              | creating                             |
| type                | default                              |
| updated_at          | None                                 |
| user_id             | 51ce99c11f9e4ed08e92acca176c33ca     |
+---------------------+--------------------------------------+
```

You must now wait until the volume status changes from `creating` to `available`. One way to do this is with a bash `until` loop:

```
until [ `openstack volume show -f value -c status targetvol` = "available" ]; do
  sleep 5
done
```

Retyping the target volume (optional)

If you want to retain the current type of your target volume, you can safely skip this step.

If you do need to select a different volume type for your target volume (see the relevant how-to guide for details on retyping), now is the time to do so. Set the new volume type, and then wait for the retype operation to complete.

```
$ openstack volume set --type <new-type> --retype-policy on-demand targetvol

$ until [ `openstack volume show -f value -c status targetvol` = "available" ]; do
  sleep 5
done
```

Attaching the target volume

You can now attach the target volume to your server:

```
$ openstack server add volume testsrv targetvol
+----------------------+--------------------------------------+
| Field                | Value                                |
+----------------------+--------------------------------------+
| ID                   | d2a22868-a133-40a1-b1a6-0cbae3feaf8d |
| Server ID            | 23a391f7-57ba-4c7f-bdd1-d1b89d6e39b2 |
| Volume ID            | e233e7f3-f33b-4d7a-8f5b-785b34f670bf |
| Device               | /dev/vdc                             |
| Tag                  | None                                 |
| Delete On Termination | False                               |
+----------------------+--------------------------------------+
```

Synchronizing data between the source and target volume

At this point, your server contains *current* data on the source volume (`sourcevol`), and *outdated* data on the target volume (`targetvol`), since your application has continued to write data since you took the snapshot.

Thus, you must now conduct a final synchronization of your data. How you do this precisely depends on your workload, but certain rules of thumb apply based on the guest operating system.

Linux/BSD/Unix **Windows**

1. Mount the device corresponding to the target volume to a temporary path. This may entail that you make some modifications to the filesystem prior to mounting. For example, an XFS filesystem will need a new UUID, which you can set with `xfs_admin -U generate <device>`
2. Synchronize your data between the source volume's mount point and the target volume's temporary one, for example:

   ```
   rsync -av /srv/data /mnt
   ```

   You can repeat this step as often as necessary.
3. Stop any services accessing data on the source volume (this marks the start of your migration downtime).
4. Run a final synchronization:

   ```
   rsync -av /srv/data /mnt
   ```

5. Unmount the source volume.
6. Remount the target volume to the source volume's prior mount point.
7. Start the service accessing data on the target volume (this marks the end of your migration downtime).

1. Assign a drive letter to the device corresponding to the target volume (this example assumes `E:`)
2. Synchronize your data between the source volume's drive letter (this example assumes `D:`) and the target volume's temporary one, for example:

   ```
   %SystemRoot%\system32\robocopy.exe
   D: E: /MT:16 /R:0 /W:0 /ZB /NP /
   ```

```
COPYALL /DCOPY:T /MIR /NFL /NDL /
XJD /XO
```

You can repeat this step as often as necessary.

3. Stop any services accessing data on the source volume (this marks the start of your migration downtime).
4. Run a final synchronization:

```
%SystemRoot%\system32\robocopy.exe
D: E: /MT:16 /R:0 /W:0 /ZB /NP /
COPYALL /DCOPY:T /MIR /NFL /NDL /
XJD /XO
```

5. Unassign the drive letter (`D:`) from the device corresponding to the source volume.
6. Change the drive letter of the device corresponding to the target volume (`E:`) to that previously used by the device corresponding to the source
7. Start the service accessing data on the target volume (this marks the end of your migration downtime).

Detaching the source volume

Finally, detach the source volume from the server.

```
$ openstack server remove volume testsrv sourcevol
```

Marking the source volume read-only (optional)

If you do not want to delete the source volume straight away, but retain it as a backup in case anything has gone wrong in the migration, it makes good sense to mark it as read-only. That way, if the source volume is accidentally attached to a server, its data cannot be modified.

```
$ openstack volume set --read-only sourcevol
$ openstack volume show sourcevol
+------------------------------+--------------------------------------+
| Field                        | Value                                |
+------------------------------+--------------------------------------+
| attachments                  | []                                   |
| availability_zone            | nova                                 |
| bootable                     | false                                |
| consistencygroup_id          | None                                 |
| created_at                   | 2023-01-05T14:00:40.000000           |
| description                  | None                                 |
| encrypted                    | False                                |
| id                           | 526f7741-2c44-4d04-a0c8-86b8d039e674 |
| multiattach                  | False                                |
```

```
| name                         | sourcevol                          |
| properties                   | readonly='True'                    |
| replication_status           | None                               |
| size                         | 50                                 |
| snapshot_id                  | None                               |
| source_volid                 | None                               |
| status                       | available                          |
| type                         | default                            |
| updated_at                   | 2023-01-05T16:03:55.000000         |
| user_id                      | 51ce99c11f9e4ed08e92acca176c33ca   |
+------------------------------+------------------------------------+
```

Last update: 2023-01-05

Created: 2023-01-05

Authors: Florian Haas