Work with volumes
Astra Trident
NetApp
March 05, 2024
# Table of Contents

Work with volumes ............................................................................................................................................... 1  
  Create a volume ............................................................................................................................................... 1  
  Remove a volume ............................................................................................................................................ 1  
  Clone a volume ............................................................................................................................................... 1  
Access externally created volumes .................................................................................................................. 3  
Driver-specific volume options .......................................................................................................................... 3
Work with volumes

You can easily create, clone, and remove volumes using the standard `docker volume` commands with the Astra Trident driver name specified when needed.

Create a volume

- Create a volume with a driver using the default name:
  
  ```bash
  docker volume create -d netapp --name firstVolume
  ```

- Create a volume with a specific Astra Trident instance:
  
  ```bash
  docker volume create -d ntap_bronze --name bronzeVolume
  ```

  If you do not specify any options, the defaults for the driver are used.

- Override the default volume size. See the following example to create a 20GiB volume with a driver:
  
  ```bash
  docker volume create -d netapp --name my_vol --opt size=20G
  ```

  Volume sizes are expressed as strings containing an integer value with optional units (example: 10G, 20GB, 3TiB). If no units are specified, the default is G. Size units can be expressed either as powers of 2 (B, KiB, MiB, GiB, TiB) or powers of 10 (B, KB, MB, GB, TB). Shorthand units use powers of 2 (G = GiB, T = TiB, …).

Remove a volume

- Remove the volume just like any other Docker volume:
  
  ```bash
  docker volume rm firstVolume
  ```

  When using the `solidfire-san` driver, the above example deletes and purges the volume.

Perform the steps below to upgrade Astra Trident for Docker.

Clone a volume

When using the `ontap-nas`, `ontap-san`, `solidfire-san`, and `gcp-cvs` storage drivers, Astra Trident can clone volumes. When using the `ontap-nas-flexgroup` or `ontap-nas-economy` drivers,
cloning is not supported. Creating a new volume from an existing volume will result in a new snapshot being created.

- Inspect the volume to enumerate snapshots:

```
docker volume inspect <volume_name>
```

- Create a new volume from an existing volume. This will result in a new snapshot being created:

```
docker volume create -d <driver_name> --name <new_name> -o from=<source_docker_volume>
```

- Create a new volume from an existing snapshot on a volume. This will not create a new snapshot:

```
docker volume create -d <driver_name> --name <new_name> -o from=<source_docker_volume> -o fromSnapshot=<source_snap_name>
```

Example
```bash
docker volume inspect firstVolume

[
  {
    "Driver": "ontap-nas",
    "Labels": null,
    "Mountpoint": "/var/lib/docker-volumes/ontap-nas/netappdvp_firstVolume",
    "Name": "firstVolume",
    "Options": {},
    "Scope": "global",
    "Status": {
      "Snapshots": [
        {
          "Created": "2017-02-10T19:05:00Z",
          "Name": "hourly.2017-02-10_1505"
        }
      ]
    }
  }
]

docker volume create -d ontap-nas --name clonedVolume -o from=firstVolume
clonedVolume

docker volume rm clonedVolume

docker volume create -d ontap-nas --name volFromSnap -o from=firstVolume -o fromSnapshot=hourly.2017-02-10_1505
volFromSnap

docker volume rm volFromSnap
```

**Access externally created volumes**

You can access externally created block devices (or their clones) by containers using Trident **only** if they have no partitions and if their filesystem is supported by Astra Trident (for example: an ext4-formatted /dev/sdc1 will not be accessible via Astra Trident).

**Driver-specific volume options**

Each storage driver has a different set of options, which you can specify at volume creation time to customize the outcome. See below for options that apply to your configured storage system.

Using these options during the volume create operation is simple. Provide the option and the value using the
-o operator during the CLI operation. These override any equivalent values from the JSON configuration file.

**ONTAP volume options**

Volume create options for both NFS and iSCSI include the following:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>size</td>
<td>The size of the volume, defaults to 1 GiB.</td>
</tr>
<tr>
<td>spaceReserve</td>
<td>Thin or thick provision the volume, defaults to thin. Valid values are <code>none</code> (thin provisioned) and <code>volume</code> (thick provisioned).</td>
</tr>
<tr>
<td>snapshotPolicy</td>
<td>This will set the snapshot policy to the desired value. The default is <code>none</code>, meaning no snapshots will automatically be created for the volume. Unless modified by your storage administrator, a policy named “default” exists on all ONTAP systems which creates and retains six hourly, two daily, and two weekly snapshots. The data preserved in a snapshot can be recovered by browsing to the <code>.snapshot</code> directory in any directory in the volume.</td>
</tr>
<tr>
<td>snapshotReserve</td>
<td>This will set the snapshot reserve to the desired percentage. The default is no value, meaning ONTAP will select the snapshotReserve (usually 5%) if you have selected a snapshotPolicy, or 0% if the snapshotPolicy is none. You can set the default snapshotReserve value in the config file for all ONTAP backends, and you can use it as a volume creation option for all ONTAP backends except ontap-nas-economy.</td>
</tr>
<tr>
<td>splitOnClone</td>
<td>When cloning a volume, this will cause ONTAP to immediately split the clone from its parent. The default is <code>false</code>. Some use cases for cloning volumes are best served by splitting the clone from its parent immediately upon creation, because there is unlikely to be any opportunity for storage efficiencies. For example, cloning an empty database can offer large time savings but little storage savings, so it’s best to split the clone immediately.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>encryption</td>
<td>Enable NetApp Volume Encryption (NVE) on the new volume; defaults to false. NVE must be licensed and enabled on the cluster to use this option. If NAE is enabled on the backend, any volume provisioned in Astra Trident will be NAE enabled. For more information, refer to: How Astra Trident works with NVE and NAE.</td>
</tr>
<tr>
<td>tieringPolicy</td>
<td>Sets the tiering policy to be used for the volume. This decides whether data is moved to the cloud tier when it becomes inactive (cold).</td>
</tr>
</tbody>
</table>

The following additional options are for NFS only:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>unixPermissions</td>
<td>This controls the permission set for the volume itself. By default the permissions will be set to <code>---rwxr-xr-x</code>, or in numerical notation 0755, and root will be the owner. Either the text or numerical format will work.</td>
</tr>
<tr>
<td>snapshotDir</td>
<td>Setting this to true will make the .snapshot directory visible to clients accessing the volume. The default value is false, meaning that visibility of the .snapshot directory is disabled by default. Some images, for example the official MySQL image, don’t function as expected when the .snapshot directory is visible.</td>
</tr>
<tr>
<td>exportPolicy</td>
<td>Sets the export policy to be used for the volume. The default is default.</td>
</tr>
<tr>
<td>securityStyle</td>
<td>Sets the security style to be used for access to the volume. The default is unix. Valid values are unix and mixed.</td>
</tr>
</tbody>
</table>

The following additional options are for iSCSI only:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fileSystemType</td>
<td>Sets the file system used to format iSCSI volumes. The default is ext4. Valid values are ext3, ext4, and xfs.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>spaceAllocation</td>
<td>Setting this to false will turn off the LUN’s space-allocation feature. The default value is true, meaning ONTAP notifies the host when the volume has run out of space and the LUN in the volume cannot accept writes. This option also enables ONTAP to reclaim space automatically when your host deletes data.</td>
</tr>
</tbody>
</table>

**Examples**

See the examples below:

- Create a 10GiB volume:

  ```bash
docker volume create -d netapp --name demo -o size=10G -o encryption=true
  ```

- Create a 100GiB volume with snapshots:

  ```bash
docker volume create -d netapp --name demo -o size=100G -o snapshotPolicy=default -o snapshotReserve=10
  ```

- Create a volume which has the setUID bit enabled:

  ```bash
docker volume create -d netapp --name demo -o unixPermissions=4755
  ```

  The minimum volume size is 20MiB.

  If the snapshot reserve is not specified and the snapshot policy is none, Trident will use a snapshot reserve of 0%.

  - Create a volume with no snapshot policy and no snapshot reserve:

    ```bash
docker volume create -d netapp --name my_vol --opt snapshotPolicy=none
    ```

  - Create a volume with no snapshot policy and a custom snapshot reserve of 10%:

    ```bash
docker volume create -d netapp --name my_vol --opt snapshotPolicy=none --opt snapshotReserve=10
    ```

  - Create a volume with a snapshot policy and a custom snapshot reserve of 10%:
• Create a volume with a snapshot policy, and accept ONTAP’s default snapshot reserve (usually 5%):

    docker volume create -d netapp --name my_vol --opt snapshotPolicy=myPolicy

---

**Element software volume options**

The Element software options expose the size and quality of service (QoS) policies associated with the volume. When the volume is created, the QoS policy associated with it is specified using the `-o type=service_level` nomenclature.

The first step to defining a QoS service level with the Element driver is to create at least one type and specify the minimum, maximum, and burst IOPS associated with a name in the configuration file.

Other Element software volume create options include the following:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>size</td>
<td>The size of the volume, defaults to 1GiB or config entry … &quot;defaults&quot;: {&quot;size&quot;: &quot;5G&quot;}.</td>
</tr>
<tr>
<td>blocksize</td>
<td>Use either 512 or 4096, defaults to 512 or config entry DefaultBlockSize.</td>
</tr>
</tbody>
</table>

**Example**

See the following sample configuration file with QoS definitions:
In the above configuration, we have three policy definitions: Bronze, Silver, and Gold. These names are arbitrary.

- Create a 10GiB Gold volume:

  ```bash
docker volume create -d solidfire --name sfGold -o type=Gold -o size=10G
  ```

- Create a 100GiB Bronze volume:

  ```bash
docker volume create -d solidfire --name sfBronze -o type=Bronze -o size=100G
  ```