



SnapDrive for UNIX operations on FlexClone volumes

Snapdrive for Unix

NetApp
February 12, 2024

This PDF was generated from https://docs.netapp.com/us-en/snapdrive-unix/aix/concept_role_based_access_control_for_flexclone_volume_operations.html on February 12, 2024. Always check docs.netapp.com for the latest.

Table of Contents

- SnapDrive for UNIX operations on FlexClone volumes 1
 - Role-based access control for FlexClone volume operations 1
 - Snap connect procedure 1
 - Snap disconnect procedure 2
 - Connecting to a single filespec in a LUN 2
 - Connecting to multiple filespecs 4
 - Disconnecting a file specification 6
 - Splitting the FlexClone volume 7
 - Space reservation with a FlexClone volume 8

SnapDrive for UNIX operations on FlexClone volumes

You can carry out various operations on FlexClone volumes using SnapDrive for UNIX.

Role-based access control for FlexClone volume operations

If Operations Manager is configured with the system, the Operations Manager administrator must grant you the required capabilities to carry out storage operations on FlexClone volumes.

Unrestricted FlexClone volume: You must have the following capabilities to perform storage operations on an unrestricted FlexClone volume:

- For `snap connect`, you must have `SD.SnapShot.UnrestrictedClone` capability on the parent volume.
- For `snap disconnect`, you must have `SD.SnapShot.DestroyUnrestrictedClone` capability on the FlexClone volume.

Restricted FlexClone volumes: You must have the following capabilities for storage operations on a restricted FlexClone volume:

- For `snap connect` you must have `SD.SnapShot.Clone` capability on the parent volume.
- For `snap disconnect` you must have `SD.SnapShot.Clone` capability on the parent volume.

To split the FlexClone volume during the `snap connect` operation, `SD.Storage.Write` access capability on the parent volume.

To perform storage operations on split FlexClone volumes during the `snap disconnect`, for split FlexClone volumes, you must have `SD.Storage.Delete` capability on the split volume clone.

Snap connect procedure

The `snap connect` procedure is valid for Data ONTAP 7.2 or later and explain the steps for `snap connect` operation.

The following steps depict the `snap connect` procedure:

Steps

1. SnapDrive for UNIX obtains the file spec name and the Snapshot name from the `snap connect` CLI. It then finds out the original storage system volume(s) where the snapshot(s) resides.
2. SnapDrive for UNIX chooses an appropriate cloning method for storage system entities based on CLI configuration options.



If the CLI option is specified, it always overrides the `snapdrive.conf` file option.

- a. If `-clone lunclone` option is specified in the CLI or the `san-clone-method=lunclone` option in

the `snapdrive.conf` file, then SnapDrive for UNIX creates a LUN clone in the same volume from the given snapshot.

- b. If `-clone unrestricted` option is specified in the CLI or the `san-clone-method=unrestricted` option in the `snapdrive.conf` file, then SnapDrive for UNIX creates or reuses all-purpose FlexClone from the given Snapshot.
- c. If `-clone optimal` option is specified in the CLI or the `san-clone-method=optimal` option in the `snapdrive.conf` file, then SnapDrive for UNIX automatically chooses between restricted FlexClone and LUN clone, based on storage system configuration.



By default, clones created by SnapDrive for UNIX are backup clones.

Snap disconnect procedure

This section explains the procedure followed for snap disconnect.

The following steps depict the snap disconnect procedure:

Steps

1. SnapDrive for UNIX obtains host file specs from `snap disconnect` CLI and finds out the storage system volume.
2. After deleting host file specs and un-mapping the LUNs, SnapDrive for UNIX checks if the current volume is a clone created by SnapDrive.
3. If the FlexClone is created by SnapDrive, then SnapDrive for UNIX checks whether:
 - a. There are any mapped LUNs in the clone volume
 - b. There are any new LUNs in the cloned volume
 - c. There are any new Snapshot copies

If all the conditions are satisfied, SnapDrive for UNIX deletes the cloned volume. For an unrestricted FlexClone, if RBAC is configured, then SnapDrive for UNIX does not perform substep 1 and substep 2 checks.

On storage systems with Data ONTAP version earlier than 7.2, restricted FlexClone volumes does not gets deleted by SnapDrive for UNIX. Storage system commands should be used to delete them. However, Unrestricted FlexClone volumes are deleted when the appropriate RBAC capability is provided to the user.

Related information

[Role-based access control in SnapDrive for UNIX](#)

Connecting to a single filespec in a LUN

You can connect to a single filespec which resides in a LUN. The following example describes how you can complete this operation.



Filespec can be any one of file system, host volume, disk group, or a LUN.

For creating restricted FlexClone, you need to set the option `san-clone-method =optimal` in `snapdrive.conf` file or give `-clone optimal` before you can connect the filespec. This example displays the results when you want to connect a filespec `/mnt/fs1` which resides in a LUN `homer:/vol/vol1/lun1`

The layout of the source file system /mnt/fs1 is given below:

```
# snapdrive storage show -fs /mnt/fs_1 -capabilities
raw device: /dev/sdc1 mount point: /mnt/fs_1 (persistent) fstype jfs2

allowed operations: all

device filename adapter path size proto state clone lun path backing
snapshot
-----
/dev/sdc - P 100m iscsi online No homer:/vol/vol1/lun1 -
```

```
# snapdrive snap connect -fs /mnt/fs_1 -snapname homer:/vol/vol1:snap1
-autorename

connecting /mnt/fs_1:
    creating restricted volume clone
homer:/vol/Snapdrive_vol1_volume_clone_from_snap1_snapshot ... success

    mapping new lun(s) ... done
    discovering new lun(s) ... done
Successfully connected to snapshot homer:/vol/vol1:snap1
    file system: /mnt/fs_1_0
```

Here, SnapDrive for UNIX automatically chose to create a restricted FlexClone

Snapdrive_vol1_0_volume_clone_from_snap1_snapshot from snapshot snap1 and connected it to the LUN testlun1 in this FlexClone to create a cloned host file system /mnt/fs_1_0.

```
# snapdrive storage show -fs /mnt/fs_1_0 -capabilities


raw device: /dev/sde1 mount point: /mnt/fs_1_0 (persistent) fstype jfs2
allowed operations: snap disconnect; snap list; snap show; storage show;
storage list

device filename adapter path size proto state clone lun path backing
snapshot
-----
/dev/sde - P 100m iscsi online flex-clone
homer:/vol/Snapdrive_vol1_volume_clone_from_snap1_snapshot/lun1 vol1:snap1
```

SnapDrive for UNIX names the destination FlexClone with the prefix mentioned in the CLI.

If the generated FlexClone name (with prefix) is already in use, SnapDrive for UNIX gives an error message. In this case, you can use `-autorename` option along with `-prefixfv` to auto generate clone volume names.

`-prefixfv` CLI option is applicable only for SAN based filespecs. The `-destfv` CLI option is ignored by SnapDrive for UNIX for SAN based filespecs during FlexClone operation.



SnapDrive for UNIX displays a warning when a Snapshot copy is locked due to LUN clones. However, SnapDrive for UNIX proceeds with the `snap connect` operation.

Connecting to multiple filespecs

You can connect to multiple filespecs that reside in a volume. The following example describes how you can complete this operation.

For creating restricted FlexClone, you need to set the `san-clone-method=unrestricted` in the `snapdrive.conf` file or give `-clone unrestricted` before you can connect to multiple filespec. This example displays the results when you want to connect to multiple filespec `/mnt/fs_1` and `/mnt/fs_2` in `snap1` which reside in volume `vol1`.

The layout of `/mnt/fs_1` and `/mnt/fs_2` is given in the following:

```
# snapdrive storage show -fs /mnt/fs_1 /mnt/fs_2 -capabilities
raw device: /dev/sdc1    mount point: /mnt/fs_1 (persistent) fstype jfs2
allowed operations: all

device filename      adapter  path      size    proto    state    clone
lun path            backing snapshot
-----
-----
/dev/sdc              -        P        100m    iscsi    online   No
homer:/vol/vol1/lun1    -

raw device: /dev/sd01    mount point: /mnt/fs_2 (persistent) fstype jfs2
allowed operations: all

device filename      adapter path      size    proto    state    clone    lun
path              backing snapshot
-----
-----
/dev/sdo            -        P        100m    iscsi    online   No
homer:/vol/vol1/lun2    -
```

Here the user used Snapshot `snap2` which contains application consistent Snapshots of `/mnt/fs1` and `/mnt/fs2` for `snap connect`.

```
# snapdrive snap connect -fs /mnt/fs_1 -fs /mnt/fs_2 -snapname
homer:/vol/vol1:snap2 -autorename -clone unrestricted

connecting /mnt/fs_1:
    creating unrestricted volume clone
homer:/vol/Snapdrive_voll_volume_clone_from_snap2_snapshot ... success

    mapping new lun(s) ... done

connecting /mnt/fs_2:

    mapping new lun(s) ... done
    discovering new lun(s) ... done
Successfully connected to snapshot homer:/vol/vol1:snap2
    file system: /mnt/fs_1_0
    file system: /mnt/fs_2_0
```

Here, SnapDrive for UNIX has created a single unrestricted FlexClone

`Snapdrive_voll_0_volume_clone_from_snap2_Snapshot` from Snapshot `snap2`. This is then connected to LUN `testlun1` and `testlun2` to create a cloned host file system `/mnt/fs1_0` and `/mnt/fs2_0` respectively.

```
# snapdrive storage show -fs /mnt/fs_1_0 /mnt/fs_2_0 -capabilities
raw device: /dev/sde1    mount point: /mnt/fs_1_0 (persistent) fstype jfs2
allowed operations: all

device filename          adapter  path  size  proto  state  clone
lun path                                     backing
snapshot
-----
-----
-----
/dev/sde                  -        P    100m  iscsi  online  flex-clone
homer:/vol/Snapdrive_voll_volume_clone_from_snap2_snapshot/lun1
voll:snap2

raw device: /dev/sdac1  mount point: /mnt/fs_2_0 (persistent) fstype jfs2
allowed operations: all

device filename          adapter  path  size  proto  state  clone
lun path
backing snapshot
-----
-----
-----
/dev/sdac                  -        P    100m  iscsi  online  flex-clone
homer:/vol/Snapdrive_voll_volume_clone_from_snap2_snapshot/lun2
voll:snap2
```

Disconnecting a file specification

You can disconnect a file specification. The following example describes how you can complete this operation.

The layout of /mnt/fs1_0 is given in the following.


```
# snapdrive storage show -fs /mnt/fs_1_0 -capabilities
raw device: /dev/sde1    mount point: /mnt/fs_1_0 (persistent) fstype jfs2
allowed operations: all
```

device	filename	adapter	path	size	proto	state	clone
lun path							backing
snapshot							
-----		-----	----	----	-----	-----	-----

/dev/sde		-	P	100m	iscsi	online	flex-clone
homer:/vol/Snapdrive_voll_volume_clone_from_snap2_snapshot/lun1							
voll:snap2							

The following output shows how to disconnect the file system /mnt/fs_1_0:

```
#snapdrive snap disconnect -fs /mnt/fs_1_0
delete file system /mnt/fs_1_0
- fs /mnt/fs_1_0 ... deleted
    - LUN
homer:/vol/Snapdrive_voll_volume_clone_from_snap2_snapshot/lun1 ...
disconnected
    - deleting volume clone ...
homer:/vol/Snapdrive_voll_volume_clone_from_snap2_snapshot done
```

After disconnecting the file system /mnt/fs_1_0, SnapDrive for UNIX also deletes the cloned volume Snapdrive_voll_0_volume_clone_from_snap2_snapshot, if the following conditions are met:

- No mapped LUNs are present in the volume
- No new LUNs are present
- No new Snapshot copies are present

Deleting a FlexClone not created by SnapDrive for UNIX: SnapDrive for UNIX does not allow deletion of FlexClone not created by it.

Splitting the FlexClone volume

You can split the existing FlexClone volume using the SnapDrive for UNIX commands.

To split the FlexClone volume you need to additionally specify the `-split` CLI option with the `snap connect` command. To disconnect, you should provide the `-split` option with the `disconnect` command to disconnect the split FlexClone volume. The blocking and non-blocking modes of initiating a `-split` operation are valid here.

Related information

Space reservation with a FlexClone volume

You can carry out space reservation operation on both restricted and unrestricted type of FlexClone volumes. By default FlexClone volumes do not have write reservations.

Following are the space reservation settings for the FlexClone for unrestricted and restricted FlexClone volumes:

Unrestricted FlexClone volumes: If `-reserve` option is given in the CLI or `flexclone-writereserve-enabled = on` in `snapdrive.conf`, the space reservation would be set to volume on the FlexClone, otherwise it is none. SnapDrive for UNIX shows an error message if the space available on the aggregate is not sufficient to adhere to this setting.

Restricted FlexClone volumes:

- The space reservation setting at volume-level is always disabled; that is, space reservation is set to NONE.
- If `-reserve` or `-noreserve` option is given in the CLI, then space reservation is enabled or disabled respectively on the LUNs which belong to the host file spec. Otherwise, the host file spec related LUNs in FlexClone inherit the space reservation from the parent volume LUNs.

Copyright information

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

Trademark information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.