∩ NetApp

Process to create a FlexCache volume
ONTAP 9

NetApp
February 26, 2022

Table of Contents

Process to create a FlexCache volume

Create a FlexCache volume

You can create a FlexCache volume in the same cluster for improving performance when accessing a hot object. If you have data centers in different locations, you can create FlexCache volumes on remote clusters for accelerating data access.

About this task

The FlexCache volume is always a FlexGroup volume, and not a FlexVol volume.

Beginning withONTAP 9.7, FlexGroup volumes are also supported at the origin of the FlexCache relationship.

Steps

1. If the FlexCache volume to be created is in a different cluster, create a cluster peer relationship:

   a. On the destination cluster, create a peer relationship with the data protection source cluster:
      ```
      cluster peer create -generate-passphrase -offer-expiration MM/DD/YYYY
      HH:MM:SS|1…7days|1…168hours -peer-addrs peer_LIF_IPs -initial-allowed
      -vserver-peers svm_name,..|* -ipspace ipspace_name
      ```

      Beginning with ONTAP 9.6, TLS encryption is enabled by default when creating a cluster peer relationship. TLS encryption is supported for the intercluster communication between the origin and FlexCache volumes. You can also disable TLS encryption for the cluster peer relationship, if required.

      ```
      cluster02::> cluster peer create -generate-passphrase -offer
      -expiration 2days -initial-allowed-vserver-peers *

                          Passphrase: UCa+6lRVICXeL/gq1WrK7ShR
                     Expiration Time: 6/7/2017 08:16:10 EST
        Initial Allowed Vserver Peers: *
                  Intercluster LIF IP: 192.140.112.101
                    Peer Cluster Name: Clus_7ShR (temporary generated)

      Warning: make a note of the passphrase - it cannot be displayed
      again.
      ```

   b. On the source cluster, authenticate the source cluster to the destination cluster:
      ```
      cluster peer create -peer-addrs peer_LIF_IPs -ipspace ipspace
      ```

```
cluster01::> cluster peer create -peer-addrs
192.140.112.101,192.140.112.102

Notice: Use a generated passphrase or choose a passphrase of 8 or
more characters.
        To ensure the authenticity of the peering relationship, use a
phrase or sequence of characters that would be hard to guess.

Enter the passphrase:
Confirm the passphrase:

Clusters cluster02 and cluster01 are peered.
```

2. If the FlexCache volume is in a different SVM than that of the origin volume, create an SVM peer relationship with `flexcache` as the application:

   a. If the SVM is in a different cluster, create an SVM permission for the peering SVMs:

      ```
      vserver peer permission create -peer-cluster cluster_name -vserver svm-name
      -applications flexcache
      ```

      The following example illustrates how to create an SVM peer permission that applies for all of the local SVMs:

      ```
      cluster1::> vserver peer permission create -peer-cluster cluster2
      -vserver "*" -applications flexcache

      Warning: This Vserver peer permission applies to all local Vservers.
      After that no explict
      "vserver peer accept" command required for Vserver peer relationship
      creation request
      from peer cluster "cluster2" with any of the local Vservers. Do you
      want to continue? {y|n}: y
      ```

   b. Create the SVM peer relationship:

      ```
      vserver peer create -vserver local_SVM -peer-vserver remote_SVM -peer
      cluster cluster_name -applications flexcache
      ```

3. Create a FlexCache volume:

   ```
   volume flexcache create -vserver cache_svm -volume cache_vol_name -auto
   -provision-as flexgroup -size vol_size -origin-vserver origin_svm -origin
   -volume origin_vol_name
   ```

   The following example creates a FlexCache volume and automatically selects existing aggregates for provisioning:

```
cluster1::> volume flexcache create -vserver vs_1 -volume fc1 -auto
-provision-as flexgroup -origin-volume vol_1 -size 160MB -origin-vserver
vs_1
[Job 443] Job succeeded: Successful
```

The following example creates a FlexCache volume and sets the junction path:

```
cluster1::> flexcache create -vserver vs34 -volume fc4 -aggr-list
aggr34,aggr43 -origin-volume origin1 -size 400m -junction-path /fc4
[Job 903] Job succeeded: Successful
```

4. Verify the FlexCache relationship from the FlexCache volume and the origin volume.

   a. View the FlexCache relationship in the cluster:
      ```
      volume flexcache show
      ```

      ```
      cluster1::> volume flexcache show
      Vserver Volume        Size         Origin-Vserver Origin-Volume Origin-
      Cluster
      ------- ----------- ---------- -------------- -------------
      -------------
      vs_1    fc1           160MB        vs_1           vol_1
      cluster1
      ```

   b. View all of the FlexCache relationships in the origin cluster:
      ```
      volume flexcache origin show-caches
      ```

      ```
      cluster::> volume flexcache origin show-caches
      Origin-Vserver Origin-Volume   Cache-Vserver   Cache-Volume   Cache-
      Cluster
      ------------- -------------- -------------- -------------
      --------------
      vs0            ovol1          vs1             cfg1           clusA
      vs0            ovol1          vs2             cfg2           clusB
      vs_1           vol_1          vs_1            fc1
      cluster1
      ```

Result

The FlexCache volume is successfully created. Clients can mount the volume by using the junction path of the FlexCache volume.

Related information

Guidelines for sizing a FlexCache volume

You must be aware of the limits for FlexCache volumes before you start provisioning the volumes.

The size limit of a FlexVol volume is applicable to an origin volume. The size of a FlexCache volume can be less than or equal to the origin volume. The best practice for the size of a FlexCache volume is to be at least 10 percent of the size of the origin volume.

You must also be aware of the following additional limits on FlexCache volumes:

Limit	ONTAP 9.5-9.6	ONTAP 9.7	ONTAP 9.8 and later
Maximum number of FlexCache volumes that you can create from an origin volume	10	10	100
Recommended maximum number of origin volumes per node	10	100	100
Recommended maximum number of FlexCache volumes per node	10	100	100
Recommended maximum number of FlexGroup constituents in a FlexCache volume per node	40	800	800
Maximum number of constituents per FlexCache volume per node	32	32	32

Related information

NetApp Interoperability

Considerations for auditing FlexCache volumes

Beginning with ONTAP 9.7, you can audit NFS file access events in FlexCache relationships using native ONTAP auditing and file policy management with FPolicy. FPolicy is not supported for FlexCache volumes with SMB. Native auditing and FPolicy are configured and managed with the same CLI commands used for FlexVol volumes. However, there is some different behavior with FlexCache volumes.

- **Native auditing**
  - You can't use a FlexCache volume as the destination for audit logs.
  - If you want to audit read and writes on FlexCache volumes, you must configure auditing on both the cache SVM as well as on the origin SVM.

    This is because file system operations are audited where they are processed. That is, reads are audited on the cache SVM and writes are audited on the origin SVM.

- To track the origin of write operations, the SVM UUID and MSID are appended in the audit log to identify the FlexCache volume from which the write originated.
- Although system access control lists (SACLs) can be set on a file using NFSv4 or SMB protocols, FlexCache volumes support only NFSv3. Therefore, SACLs can only be set on the origin volume.

- **FPolicy**
  - Although writes to a FlexCache volume are committed on the origin volume, FPolicy configurations monitor the writes on the cache volume. This is unlike native auditing, in which the writes are audited on the origin volume.
  - While ONTAP does not require the same FPolicy configuration on cache and origin SVMs, it is recommended that you deploy two similar configurations. You can do so by creating a new FPolicy policy for the cache, configured like that of the origin SVM but with the scope of the new policy limited to the cache SVM.