vserver locks commands

ONTAP commands

NetApp
February 04, 2026

Table of Contents

vserver locks commands

vserver locks break

Break file locks based on a set of criteria

Availability: This command is available to *cluster* and *Vserver* administrators at the *advanced* privilege level.

Description

The `vserver locks break` command breaks one or more locks.

Parameters

{ `-vserver <vserver name>` - Vserver (privilege: advanced)

This parameter specifies the Vserver containing the lock.

`-volume <volume name>` - Volume (privilege: advanced)

This parameter specifies the name of the volume containing the lock.

`-lif <lif-name>` - Logical Interface (privilege: advanced)

This parameter specifies the logical interface through which the lock was established.

`-path <text>` - Object Path (privilege: advanced)

This parameter specifies a path to the lock.

| `-lockid <UUID>` - Lock UUID (privilege: advanced) }

This parameter specifies the universally unique identifier (UUID) for the lock. Queries and wildcard characters are not supported.

`-owner-id <text>` - Owner ID (privilege: advanced)

This parameter specifies an owner ID for a lock. This parameter must be used with the query notation { } exhibited in the second example.

`-protocol <lock protocol>` - Lock Protocol (privilege: advanced)

This parameter specifies the protocol that was used to establish a lock. This parameter must be used with the query notation { } exhibited in the second example.

`-client-address <IP Address>` - Client Address (privilege: advanced)

This parameter specifies a client address associated with a lock. This parameter must be used with the query notation { } exhibited in the second example.

`-client-address-type {ipv4|ipv6|ipv6z}` - Client Address Type (privilege: advanced)

This parameter specifies the type of ip address a client used to create its lock (ipv4, ipv6). This parameter must be used with the query notation { } exhibited in the second example.

Examples

The following example breaks the locks on all objects on the Vserver named vs0 in the volume named vol0,

regardless of the paths to the locked objects and the logical interface through which the locks were established.

```
cluster1::*> vserver locks break -vserver vs0 -volume vol0 -path * -lif *
WARNING: Breaking file locks can cause applications to become
unsynchronized
        and may lead to data corruption.
Do you want to continue? {y|n}: y
1 entry was acted on.
```

The vserver locks break command can also be issued using a query on the parameters available to the vserver locks show command. The following example breaks all NLM protocol lock objects locked by the client at address 12.34.56.78.

```
cluster1::*> vserver locks break { -protocol nlm -client-address
12.34.56.78 }
Warning: Breaking file locks can cause applications to become
unsynchronized
        and may lead to data corruption.
Do you want to continue? {y|n}: y
1 entry was acted on.
```

vserver locks show

Display current list of locks

Availability: This command is available to *cluster* and *Vserver* administrators at the *admin* privilege level.

Description

The `vserver locks show` command displays information about locks. A lock is a synchronization mechanism for enforcing limits on concurrent access to files where many clients can be accessing the same file at the same time. The command output depends on the parameter or parameters specified with the command. If you do not specify any parameters, the command displays the following information about locks:

- Vserver name
- Volume name
- Object path
- Logical interface name
- Lock protocol
- Lock type
- Client

Parameters

{ [-fields <fieldname>,…]

If you specify the `-fields <fieldname>,` … parameter, the command output also includes the specified field or fields. You can use '-fields ?' to display the fields to specify.

| [-smb-attrs]

If you specify the `-smb-attrs` parameter, the command displays information related to SMB2 and higher.

| [-instance] }

If you specify the `-instance` parameter, the command displays detailed information about all fields.

{ [-vserver <vserver name>] - Vserver

If you specify this parameter, the command displays information about locks on the specified Vserver.

[-volume <volume name>] - Volume

If you specify this parameter, the command displays information about locks on the specified volume.

[-lif <lif-name>] - Logical Interface

If you specify this parameter, the command displays information about locks established through the specified logical interface.

[-path <text>] - Object Path

If you specify this parameter, the command displays information about locks at the specified path name.

| [-lockid <UUID>] - Lock UUID }

If you specify this parameter, the command displays information about the lock with the specified universally unique identifier (UUID).

[-is-constituent {true|false}] - Is Constituent Volume

If you specify this parameter, the command displays information only about volumes that either are or are not constituents of a FlexGroup, depending on the value provided. This field is *false* for constituents of an Infinite Volume.

[-protocol <lock protocol>] - Lock Protocol

If you specify this parameter, the command displays information about locks established through the specified protocol. Some of the valid protocols are:

- *cifs* : SMB locks

- *nlm* : NFS3 locks

- *nfsv4* : NFS4.0 locks

- *nfsv4.1* : NFS4.1 locks

- *crposix* : CrPosix locks for CREATE and LINK

- *fcache* : Delegations for 7-mode destination FlexCache volumes

[-type {byte-range|share-level|op-lock|delegation}] - Lock Type

If you specify this parameter, the command displays information about locks of the specified lock type. The

four types of locks are:

- Byte-range locks: Lock only a portion of a file.
- Share locks: Represent opened files.
- Opportunistic locks: Control client-side caching over SMB.
- Delegations: Control client-side caching over NFSv4.

[-node <nodename>] - Node Holding Lock State

If you specify this parameter, the command displays information about all locks on the specified node.

[-lock-state <lock_state>] - Lock State

If you specify this parameter, the command displays information about the state of the lock. Some of the valid states are:

- *granted* : The lock is established.
- *revoking* : The server is currently coordinating with the client to change the state of this lock.
- *revoked* : The lock is undergoing revocation to be downgraded or released.
- *adjusted* : The lock is undergoing revocation to be replaced by a lock equal to or weaker than the current lock.
- *subsumed* : The lock is one of a set of locks that will replace a lock that is being revoked.
- *waiting* : The lock is waiting to be granted, because it conflicts with another lock.
- *denied* : The lock has been denied.
- *timeout* : The lock was waiting and has now timed out.
- *gone* : The lock is about to be released.
- *unused* : The lock is allocated but has not been processed into any state.

[-bytelock-offset <integer>] - Bytelock Starting Offset

If you specify this parameter, the command displays information about bytelocks with the specified offset value. This is the index in the file (in bytes) where the lock begins.

[-bytelock-length <integer>] - Number of Bytes Locked

If you specify this parameter, the command displays information about bytelocks with the specified length. This is the number of bytes that are locked by this particular lock.

[-bytelock-mandatory {true|false}] - Bytelock is Mandatory

If you specify this parameter, the command displays information only about mandatory bytelocks. A mandatory bytelock enforces the requirement of byte range locking on clients before accessing the associated range.

[-bytelock-exclusive {true|false}] - Bytelock is Exclusive

If you specify this parameter, the command displays information only about exclusive bytelocks. When an exclusive bytelock is granted, no other bytelock may be granted whose range overlaps it.

[-bytelock-super {true|false}] - Bytelock is Superlock

If you specify this parameter, the command displays information only about super-bytelocks. When a super-

bytelock is granted, all other locks on that file are released, and no other operations will be allowed on that file.

[-bytelock-soft {true|false}] - Bytelock is Soft

If you specify this parameter, the command displays information only about softened bytelocks. An NFSv4 bytelock might become softened if the connection to the client is interrupted. Soft locks might be reclaimed if the client reconnects. However if another client requests a lock that conflicts with a soft lock, then the soft lock will be released.

[-oplock-level {exclusive|level2|batch|null|read-batch}] - Oplock Level

If you specify this parameter, the command displays information about locks with the specified oplock level. The oplock level determines which operations the client may cache locally. Those operations include opening, reading, writing, closing, and creating and destroying bytelocks on a file. The five valid oplock levels are:

- *batch* : The client may cache all operations on the file.

- *exclusive* : The client may cache reads and writes on the file.

- *read-batch* : The client may cache reads and opens on the file.

- *level2* : The client may cache reads on the file.

- *null* : The client may not cache any operations on the file.

[-sharelock-mode <share lock mode>] - Shared Lock Access Mode

If you specify this parameter, the command displays information about locks with the specified sharelock mode. The parameter has two components separated by a hyphen: the access mode followed by the share mode. The access mode specifies which operations the client is allowed to perform on the file. The share mode specifies which operations other clients are disallowed to perform. The two modes are a combination of one or more of these permissions:

- *read*

- *write*

- *delete*

- *all*

- *none*

For example, the sharelock mode *read_write-deny_delete* allows the client to read and write the file, and disallows other clients to delete the file. A special mode is *delete-on-close* , which specifies that the server will delete the file as soon as it is closed.

[-sharelock-soft {true|false}] - Shared Lock is Soft

If you specify this parameter, the command displays information only about softened sharelocks. A NFSv4 sharelock can become softened when the connection to the client is interrupted. If the client reconnects, it might reclaim the sharelock. However, if another client creates a sharelock that conflicts with the softened sharelock, the softened sharelock will be released.

[-delegation-type {read|write}] - Delegation Type

If you specify this parameter, the command displays information only about locks with the specified delegation-type setting. The delegation type determines which operations the client may cache locally. The two valid delegation types are:

- *read* : The client may cache reads on the file.

- *write* : The client may cache reads and writes on the file.

[-owner-id <text>] - Owner ID

If you specify this parameter, the command displays information only about locks with the specified owner ID. The owner ID is an opaque byte string generated by the server for each file lock request.

[-client-address <IP Address>] - Client Address

If you specify this parameter, the command displays information only about locks from the specified client IP address.

[-client-address-type {ipv4|ipv6|ipv6z}] - Client Address Type

If you specify this parameter, the command displays information only about locks corresponding to a certain IP address type. Please note that locks created over the NFSv4 or NFSv4.1 protocol cannot have their address types resolved. Valid options are:

- *ipv4* : Clients operating over an IPv4 interface.

- *ipv6* : Clients operating over an IPv6 interface.

[-smb-open-type {none|durable|persistent}] - SMB Open Type

If you specify this parameter, the command displays information only about locks with the specified SMB open type. Valid open types are

- *durable* : Durability is a feature of SMB2. A durable lock might become "disconnected" if the connection between the client and server is disrupted. A disconnected durable lock might be reconnected if the connection is reestablished.

- *persistent* : Persistence is a feature of SMB3. Persistent locks can become disconnected and later reconnected, like durable locks. Persistent locks are used to facilitate continuously available shares.

- *none* : The lock is neither durable nor persistent.

[-smb-connect-state <Lock Connect State>] - SMB Connect State

If you specify this parameter, the command displays information only about locks with the specified SMB connection state. Some of the valid states are:

- *connected* : This is the normal state of a SMB lock when the server and client are connected.

- *disconnected* : If a lock is durable or persistent, it might become disconnected if the connection between the server and its client is interrupted. Disconnected locks may later be reconnected if the connection is reestablished.

[-smb-expiration-time <integer>] - SMB Expiration Time (Secs)

If you specify this parameter, the command displays information only about locks with the specified SMB lock expiration time. When a lock is disconnected, -smb-expiration-time shows the time remaining until the lock expires. The server releases the lock after it expires.

[-smb-open-group-id <text>] - SMB Open Group ID

If you specify this parameter, the command displays information only about locks with the specified SMB open group identifier. This is an opaque byte string provided by the client as the SMB lease key when the lock is first established.

Examples

The following example displays default information about all locks:

```
cluster1::> vserver locks show

Vserver: vs0
Volume    Object Path               LIF         Protocol  Lock Type
Client
--------  ------------------------  ----------- --------- -----------
----------
vol1    /vol1/notes.txt             node1_data1
                                                cifs      share-level
192.168.1.5
              Sharelock Mode: read_write-deny_delete
                                                          op-lock
192.168.1.5
              Oplock Level: read-batch
        /vol1/notes1.txt           node1_data1
                                                cifs      share-level
192.168.1.5
              Sharelock Mode: read_write-deny_delete
                                                          op-lock
192.168.1.5
              Oplock Level: batch
        /vol1                      node1_data2
                                                cifs      share-level
192.168.1.5
              Sharelock Mode: read-deny_delete
        /vol1/notes.txt            node1_data2
                                                cifs      share-level
192.168.1.5
              Sharelock Mode: read_write-deny_delete
                                                          op-lock
192.168.1.5
              Oplock Level: read-batch
7 entries were displayed.
```

The following example displays the SMB related information about all locks:

```
cluster1::> vserver locks show -smb-attrs

Vserver: vs0
Volume    Object Path               LIF         Protocol  Lock Type
Client
```

```
-------- ----------------------- ----------- -------- -----------
----------
vol1      /vol1/notes.txt         node1_data1
                                                cifs     share-level
192.168.1.5
 Sharelock Mode: read_write-deny_delete
 Open Type: durable     Connect State: connected     Expiration Time
(Secs): -
 Open Group ID:
625e2ff46ee5df1194ba0050569d37047058909c00000000873d210700000000
op-lock     192.168.1.5
 Oplock Level: read-batch
 Open Type: -           Connect State: connected     Expiration Time
(Secs): -
 Open Group ID:
625e2ff46ee5df1194ba0050569d37047058909c00000000873d210700000000
/vol1/notes1.txt         node1_data1
                                                cifs     share-level
192.168.1.5
 Sharelock Mode: read_write-deny_delete
 Open Type: durable     Connect State: connected     Expiration Time
(Secs): -
 Open Group ID:
625e2ff46ee5df1194ba0050569d370440fc8891000000005a3f210700000000
op-lock     192.168.1.5
 Oplock Level: batch
 Open Type: -           Connect State: connected     Expiration Time
(Secs): -
 Open Group ID:
625e2ff46ee5df1194ba0050569d370440fc8891000000005a3f210700000000
/vol1                    node1_data2
                                                cifs     share-level
192.168.1.5
 Sharelock Mode: read-deny_delete
 Open Type: none        Connect State: connected     Expiration Time
(Secs): -
 Open Group ID: -
/vol1/notes.txt          node1_data2
                                                cifs     share-level
192.168.1.5
 Sharelock Mode: read_write-deny_delete
 Open Type: durable     Connect State: connected     Expiration Time
(Secs): -
 Open Group ID:
625e2ff46ee5df1194ba0050569d370408e08d9c00000000da40210700000000
op-lock     192.168.1.5
```

```
 Oplock Level: read-batch
 Open Type: -            Connect State: connected      Expiration Time
(Secs): -
 Open Group ID:
625e2ff46ee5df1194ba0050569d370408e08d9c00000000da40210700000000


7 entries were displayed.
```

The following example displays default information about all locks in list form:

```
cluster1::> vserver locks show -instance
Vserver: vs0
                      Volume: vol1
          Logical Interface: node1_data1
                Object Path: /vol1/notes.txt
                  Lock UUID: 447db184-f801-11df-8bb5-00a098000e34
              Lock Protocol: cifs
                  Lock Type: share-level
    Node Holding Lock State: node1
                 Lock State: granted
   Bytelock Starting Offset: -
     Number of Bytes Locked: -
       Bytelock is Mandatory: -
       Bytelock is Exclusive: -
       Bytelock is Superlock: -
            Bytelock is Soft: -
                Oplock Level: -
    Shared Lock Access Mode: read_write-deny_delete
        Shared Lock is Soft: false
            Delegation Type: -
             Client Address: 192.168.1.5
        Client Address Type: ipv4
              SMB Open Type: durable
          SMB Connect State: connected
 SMB Expiration Time (Secs): -
          SMB Open Group ID:
625e2ff46ee5df1194ba0050569d37047058909c00000000873d2107000000004
Vserver: vs0
                      Volume: vol1
          Logical Interface: node1_data1
                Object Path: /vol1/notes.txt
                  Lock UUID: 447db185-f801-11df-8bb5-00a098000e34
              Lock Protocol: cifs
                  Lock Type: op-lock
    Node Holding Lock State: node1
```

```
                      Lock State: granted
       Bytelock Starting Offset: -
         Number of Bytes Locked: -
            Bytelock is Mandatory: -
            Bytelock is Exclusive: -
            Bytelock is Superlock: -
                 Bytelock is Soft: -
                     Oplock Level: read-batch
         Shared Lock Access Mode: -
              Shared Lock is Soft: -
                  Delegation Type: -
                   Client Address: 192.168.1.5
              Client Address Type: ipv4
                    SMB Open Type: -
                SMB Connect State: connected
 SMB Expiration Time (Secs): -
                 SMB Open Group ID:
625e2ff46ee5df1194ba0050569d37047058909c00000000873d210700000000
Vserver: vs0
                          Volume: vol1
                Logical Interface: node1_data1
                      Object Path: /vol1/notes1.txt
                        Lock UUID: 48cee334-f801-11df-8bb5-00a098000e34
                    Lock Protocol: cifs
                        Lock Type: share-level
       Node Holding Lock State: node1
                       Lock State: granted
       Bytelock Starting Offset: -
         Number of Bytes Locked: -
            Bytelock is Mandatory: -
            Bytelock is Exclusive: -
            Bytelock is Superlock: -
                 Bytelock is Soft: -
                     Oplock Level: -
         Shared Lock Access Mode: read_write-deny_delete
              Shared Lock is Soft: false
                  Delegation Type: -
                   Client Address: 192.168.1.5
              Client Address Type: ipv4
                    SMB Open Type: durable
                SMB Connect State: connected
 SMB Expiration Time (Secs): -
                 SMB Open Group ID:
625e2ff46ee5df1194ba0050569d370440fc8891000000005a3f210700000000
3 entries were displayed.
```