Set up file access using SMB

ONTAP 9

NetApp
June 09, 2023

Table of Contents

Set up file access using SMB

Configure security styles

How security styles affect data access

What the security styles and their effects are

There are four different security styles: UNIX, NTFS, mixed, and unified. Each security style has a different effect on how permissions are handled for data. You must understand the different effects to ensure that you select the appropriate security style for your purposes.

It is important to understand that security styles do not determine what client types can or cannot access data. Security styles only determine the type of permissions ONTAP uses to control data access and what client type can modify these permissions.

For example, if a volume uses UNIX security style, SMB clients can still access data (provided that they properly authenticate and authorize) due to the multiprotocol nature of ONTAP. However, ONTAP uses UNIX permissions that only UNIX clients can modify using native tools.

Security style	Clients that can modify permissions	Permissions that clients can use	Resulting effective security style	Clients that can access files
UNIX	NFS	NFSv3 mode bits	UNIX	NFS and SMB
		NFSv4.x ACLs	UNIX	
NTFS	SMB	NTFS ACLs	NTFS	
Mixed	NFS or SMB	NFSv3 mode bits	UNIX	
		NFSv4.x ACLs	UNIX	
NTFS ACLs	NTFS	Unified	NFS or SMB	
NFSv3 mode bits	UNIX			
NFSv4.1 ACLs	UNIX	NTFS ACLs	NTFS	
Unified (For infinite volumes only, in ONTAP 9.4 and earlier releases.)	NFS or SMB	NFSv3 mode bits	Unix	
		NFSv4.1 ACLs		NTFS ACLs

FlexVol volumes support UNIX, NTFS, and mixed security styles. When the security style is mixed or unified, the effective permissions depend on the client type that last modified the permissions because users set the security style on an individual basis. If the last client that modified permissions was an NFSv3 client, the permissions are UNIX NFSv3 mode bits. If the last client was an NFSv4 client, the permissions are NFSv4

ACLs. If the last client was an SMB client, the permissions are Windows NTFS ACLs.

The unified security style is only available with infinite volumes, which are no longer supported in ONTAP 9.5 and later releases. For more information, see FlexGroup volumes management overview.

Beginning with ONTAP 9.2, the `show-effective-permissions` parameter to the `vserver security file-directory` command enables you to display effective permissions granted to a Windows or UNIX user on the specified file or folder path. In addition, the optional parameter `-share-name` enables you to display the effective share permission.

ONTAP initially sets some default file permissions. By default, the effective security style on all data in UNIX, mixed, and unified security style volumes is UNIX and the effective permissions type is UNIX mode bits (0755 unless specified otherwise) until configured by a client as allowed by the default security style. By default, the effective security style on all data in NTFS security style volumes is NTFS and has an ACL allowing full control to everyone.

Where and when to set security styles

Security styles can be set on FlexVol volumes (both root or data volumes) and qtrees. Security styles can be set manually at the time of creation, inherited automatically, or changed at a later time.

Decide which security style to use on SVMs

To help you decide which security style to use on a volume, you should consider two factors. The primary factor is the type of administrator that manages the file system. The secondary factor is the type of user or service that accesses the data on the volume.

When you configure the security style on a volume, you should consider the needs of your environment to ensure that you select the best security style and avoid issues with managing permissions. The following considerations can help you decide:

Security style	Choose if…
UNIX	• The file system is managed by a UNIX administrator. • The majority of users are NFS clients. • An application accessing the data uses a UNIX user as the service account.
NTFS	• The file system is managed by a Windows administrator. • The majority of users are SMB clients. • An application accessing the data uses a Windows user as the service account.
Mixed	The file system is managed by both UNIX and Windows administrators and users consist of both NFS and SMB clients.

How security style inheritance works

If you do not specify the security style when creating a new FlexVol volume or a qtree, it inherits its security style in different ways.

Security styles are inherited in the following manner:

- A FlexVol volume inherits the security style of the root volume of its containing SVM.
- A qtree inherits the security style of its containing FlexVol volume.
- A file or directory inherits the security style of its containing FlexVol volume or qtree.

How ONTAP preserves UNIX permissions

When files in a FlexVol volume that currently have UNIX permissions are edited and saved by Windows applications, ONTAP can preserve the UNIX permissions.

When applications on Windows clients edit and save files, they read the security properties of the file, create a new temporary file, apply those properties to the temporary file, and then give the temporary file the original file name.

When Windows clients perform a query for the security properties, they receive a constructed ACL that exactly represents the UNIX permissions. The sole purpose of this constructed ACL is to preserve the file's UNIX permissions as files are updated by Windows applications to ensure that the resulting files have the same UNIX permissions. ONTAP does not set any NTFS ACLs using the constructed ACL.

Manage UNIX permissions using the Windows Security tab

If you want to manipulate UNIX permissions of files or folders in mixed security-style volumes or qtrees on SVMs, you can use the Security tab on Windows clients. Alternatively, you can use applications that can query and set Windows ACLs.

- Modifying UNIX permissions

  You can use the Windows Security tab to view and change UNIX permissions for a mixed security-style volume or qtree. If you use the main Windows Security tab to change UNIX permissions, you must first remove the existing ACE you want to edit (this sets the mode bits to 0) before you make your changes. Alternatively, you can use the Advanced editor to change permissions.

  If mode permissions are used, you can directly change the mode permissions for the listed UID, GID, and others (everyone else with an account on the computer). For example, if the displayed UID has r-x permissions, you can change the UID permissions to rwx.

- Changing UNIX permissions to NTFS permissions

  You can use the Windows Security tab to replace UNIX security objects with Windows security objects on a mixed security-style volume or qtree where the files and folders have a UNIX effective security style.

  You must first remove all listed UNIX permission entries before you can replace them with the desired Windows User and Group objects. You can then configure NTFS-based ACLs on the Windows User and Group objects. By removing all UNIX security objects and adding only Windows Users and Groups to a file or folder in a mixed security-style volume or qtree, you change the effective security style on the file or folder from UNIX to NTFS.

When changing permissions on a folder, the default Windows behavior is to propagate these changes to all subfolders and files. Therefore, you must change the propagation choice to the desired setting if you do not want to propagate a change in security style to all child folders, subfolders, and files.

Configure security styles on SVM root volumes

You configure the storage virtual machine (SVM) root volume security style to determine the type of permissions used for data on the root volume of the SVM.

Steps

1. Use the `vserver create` command with the `-rootvolume-security-style` parameter to define the security style.

   The possible options for the root volume security style are `unix`, `ntfs`, or `mixed`.

2. Display and verify the configuration, including the root volume security style of the SVM you created: `vserver show -vserver` *`vserver_name`*

Configure security styles on FlexVol volumes

You configure the FlexVol volume security style to determine the type of permissions used for data on FlexVol volumes of the storage virtual machine (SVM).

Steps

1. Perform one of the following actions:

If the FlexVol volume…	Use the command…
Does not yet exist	`volume create` and include the `-security -style` parameter to specify the security style.
Already exists	`volume modify` and include the `-security -style` parameter to specify the security style.

   The possible options for the FlexVol volume security style are `unix`, `ntfs`, or `mixed`.

   If you do not specify a security style when creating a FlexVol volume, the volume inherits the security style of the root volume.

   For more information about the `volume create` or `volume modify` commands, see Logical storage management.

2. To display the configuration, including the security style of the FlexVol volume you created, enter the following command:

   ```
   volume show -volume volume_name -instance
   ```

Configure security styles on qtrees

You configure the qtree volume security style to determine the type of permissions used for data on qtrees.

Steps

1. Perform one of the following actions:

If the qtree…	Use the command…
Does not exist yet	`volume qtree create` and include the `-security-style` parameter to specify the security style.
Already exists	`volume qtree modify` and include the `-security-style` parameter to specify the security style.

   The possible options for the qtree security style are `unix`, `ntfs`, or `mixed`.

   If you do not specify a security style when creating a qtree, the default security style is `mixed`.

   For more information about the `volume qtree create` or `volume qtree modify` commands, see Logical storage management.

2. To display the configuration, including the security style of the qtree you created, enter the following command: `volume qtree show -qtree qtree_name -instance`

Create and manage data volumes in NAS namespaces

Create and manage data volumes in NAS namespaces overview

To manage file access in a NAS environment, you must manage data volumes and junction points on your storage virtual machine (SVM). This includes planning your namespace architecture, creating volumes with or without junction points, mounting or unmounting volumes, and displaying information about data volumes and NFS server or CIFS server namespaces.

Create data volumes with specified junction points

You can specify the junction point when you create a data volume. The resultant volume is automatically mounted at the junction point and is immediately available to configure for NAS access.

Before you begin

The aggregate in which you want to create the volume must already exist.

The following characters cannot be used in the junction path: * # " > < | ? \

In addition, the junction path length cannot be more than 255 characters.

Steps

1. Create the volume with a junction point: `volume create -vserver` *`vserver_name`* `-volume` *`volume_name`* `-aggregate` *`aggregate_name`* `-size {integer[KB|MB|GB|TB|PB]} -security-style {ntfs|unix|mixed} -junction-path` *`junction_path`*

   The junction path must start with the root (/) and can contain both directories and junctioned volumes. The junction path does not need to contain the name of the volume. Junction paths are independent of the volume name.

   Specifying a volume security style is optional. If you do not specify a security style, ONTAP creates the volume with the same security style that is applied to the root volume of the storage virtual machine (SVM). However, the root volume's security style might not be the security style you want applied to the data volume you create. The recommendation is to specify the security style when you create the volume to minimize difficult-to-troubleshoot file-access issues.

   The junction path is case insensitive; `/ENG` is the same as `/eng`. If you create a CIFS share, Windows treats the junction path as if it is case sensitive. For example, if the junction is `/ENG`, the path of a CIFS share must start with `/ENG`, not `/eng`.

   There are many optional parameters that you can use to customize a data volume. To learn more about them, see the man pages for the `volume create` command.

2. Verify that the volume was created with the desired junction point: `volume show -vserver` *`vserver_name`* `-volume` *`volume_name`* `-junction`

Example

The following example creates a volume named "home4" located on SVM vs1 that has a junction path `/eng/home`:

```
cluster1::> volume create -vserver vs1 -volume home4 -aggregate aggr1
-size 1g -junction-path /eng/home
[Job 1642] Job succeeded: Successful

cluster1::> volume show -vserver vs1 -volume home4 -junction
                  Junction                 Junction
Vserver   Volume  Active   Junction Path   Path Source
--------- ------- -------- --------------- -----------
vs1       home4   true     /eng/home       RW_volume
```

Create data volumes without specifying junction points

You can create a data volume without specifying a junction point. The resultant volume is not automatically mounted, and is not available to configure for NAS access. You must mount the volume before you can configure SMB shares or NFS exports for that volume.

Before you begin

The aggregate in which you want to create the volume must already exist.

Steps

1. Create the volume without a junction point by using the following command: `volume create -vserver` *`vserver_name`* `-volume` *`volume_name`* `-aggregate` *`aggregate_name`* `-size {integer[KB|MB|GB|TB|PB]} -security-style {ntfs|unix|mixed}`

   Specifying a volume security style is optional. If you do not specify a security style, ONTAP creates the volume with the same security style that is applied to the root volume of the storage virtual machine (SVM). However, the root volume's security style might not be the security style you want applied to the data volume. The recommendation is to specify the security style when you create the volume to minimize difficult-to-troubleshoot file-access issues.

   There are many optional parameters that you can use to customize a data volume. To learn more about them, see the man pages for the `volume create` command.

2. Verify that the volume was created without a junction point: `volume show -vserver` *`vserver_name`* `-volume` *`volume_name`* `-junction`

Example

The following example creates a volume named "sales" located on SVM vs1 that is not mounted at a junction point:

```
cluster1::> volume create -vserver vs1 -volume sales -aggregate aggr3
-size 20GB
[Job 3406] Job succeeded: Successful

cluster1::> volume show -vserver vs1 -junction
                     Junction                 Junction
Vserver   Volume     Active   Junction Path   Path Source
--------- ---------- -------- --------------- -----------
vs1       data       true     /data           RW_volume
vs1       home4      true     /eng/home       RW_volume
vs1       vs1_root   -        /               -
vs1       sales      -        -               -
```

Mount or unmount existing volumes in the NAS namespace

A volume must be mounted on the NAS namespace before you can configure NAS client access to data contained in the storage virtual machine (SVM) volumes. You can mount a volume to a junction point if it is not currently mounted. You can also unmount volumes.

About this task

If you unmount and offline a volume, all data within the junction point, including data in volumes with junction points contained within the unmounted volume's namespace, are inaccessible to NAS clients.

To discontinue NAS client access to a volume, it is not sufficient to simply unmount the volume. You must offline the volume, or take other steps to ensure that client-side file handle caches are invalidated. For more information, see the following Knowledge Base article:https://kb.netapp.com/Advice_and_Troubleshooting/Data_Storage_Software/ONTAP_OS/NFSv3_clients_still_have_access_to_a_volume_after_being_removed_from_the_namespace_in_ONTAP[NFSv3 clients still have access to a volume after being removed from the namespace in ONTAP]

When you unmount and offline a volume, data within the volume is not lost. Additionally, existing volume export policies and SMB shares created on the volume or on directories and junction points within the unmounted volume are retained. If you remount the unmounted volume, NAS clients can access the data contained within the volume using existing export policies and SMB shares.

Steps

1. Perform the desired action:

If you want to…	Enter the commands…
Mount a volume	`volume mount -vserver` *`svm_name`* `-volume` *`volume_name`* `-junction-path` *`junction_path`*
Unmount a volume	`volume unmount -vserver` *`svm_name`* `-volume` *`volume_name`* `volume offline -vserver` *`svm_name`* `-volume` *`volume_name`*

2. Verify that the volume is in the desired mount state: `volume show -vserver` *`vserver_name`* `-volume` *`volume_name`* `-fields state,junction-path,junction-active`

Examples

The following example mounts a volume named “sales” located on SVM vs1 to the junction point `/sales`:

```
cluster1::> volume mount -vserver vs1 -volume sales -junction-path /sales

cluster1::> volume show -vserver vs1 state,junction-path,junction-active

vserver   volume     state     junction-path   junction-active
--------- ---------- -------- --------------- ----------------
vs1       data       online    /data           true
vs1       home4      online    /eng/home       true
vs1       sales      online    /sales          true
```

The following example unmounts and offlines a volume named “data” located on SVM vs1:

```
cluster1::> volume unmount -vserver vs1 -volume data
cluster1::> volume offline -vserver vs1 -volume data

cluster1::> volume show -vserver vs1 -fields state,junction-path,junction-
active

vserver   volume     state     junction-path   junction-active
--------- ---------- --------- --------------- ---------------
vs1       data       offline   -               -
vs1       home4      online    /eng/home       true
vs1       sales      online    /sales          true
```

Display volume mount and junction point information

You can display information about mounted volumes for storage virtual machines (SVMs) and the junction points to which the volumes are mounted. You can also determine which volumes are not mounted to a junction point. You can use this information to understand and manage your SVM namespace.

Steps

1. Perform the desired action:

If you want to display…	Enter the command…
Summary information about mounted and unmounted volumes on the SVM	`volume show -vserver vserver_name -junction`
Detailed information about mounted and unmounted volumes on the SVM	`volume show -vserver vserver_name -volume volume_name -instance`
Specific information about mounted and unmounted volumes on the SVM	a. If necessary, you can display valid fields for the `-fields` parameter by using the following command: `volume show -fields ?` b. Display the desired information by using the `-fields` parameter: volume show -vserver vserver_name -fields fieldname,…

Examples

The following example displays a summary of mounted and unmounted volumes on SVM vs1:

```
cluster1::> volume show -vserver vs1 -junction
                     Junction                 Junction
Vserver   Volume     Active   Junction Path   Path Source
--------- ---------- -------- --------------- -----------
vs1       data       true     /data           RW_volume
vs1       home4      true     /eng/home       RW_volume
vs1       vs1_root   -        /               -
vs1       sales      true     /sales          RW_volume
```

The following example displays information about specified fields for volumes located on SVM vs2:

```
cluster1::> volume show -vserver vs2 -fields
vserver,volume,aggregate,size,state,type,security-style,junction-
path,junction-parent,node
vserver volume   aggregate size state  type security-style junction-path
junction-parent node
------- ------   --------- ---- ------ ---- -------------- -------------
--------------- -----
vs2     data1    aggr3     2GB  online RW   unix           -             -
node3
vs2     data2    aggr3     1GB  online RW   ntfs           /data2
vs2_root        node3
vs2     data2_1  aggr3     8GB  online RW   ntfs           /data2/d2_1
data2           node3
vs2     data2_2  aggr3     8GB  online RW   ntfs           /data2/d2_2
data2           node3
vs2     pubs     aggr1     1GB  online RW   unix           /publications
vs2_root        node1
vs2     images   aggr3     2TB  online RW   ntfs           /images
vs2_root        node3
vs2     logs     aggr1     1GB  online RW   unix           /logs
vs2_root        node1
vs2     vs2_root aggr3     1GB  online RW   ntfs           /             -
node3
```

Configure name mappings

Configure name mappings overview

ONTAP uses name mapping to map CIFS identities to UNIX identities, Kerberos identities to UNIX identities, and UNIX identities to CIFS identities. It needs this information to obtain user credentials and provide proper file access regardless of whether they are connecting from an NFS client or a CIFS client.

There are two exceptions where you do not have to use name mapping:

- You configure a pure UNIX environment and do not plan to use CIFS access or NTFS security style on volumes.
- You configure the default user to be used instead.

  In this scenario, name mapping is not required because instead of mapping every individual client credential all client credentials are mapped to the same default user.

Note that you can use name mapping only for users, not for groups.

However, you can map a group of individual users to a specific user. For example, you can map all AD users that start or end with the word SALES to a specific UNIX user and to the user's UID.

How name mapping works

When ONTAP has to map credentials for a user, it first checks the local name mapping database and LDAP server for an existing mapping. Whether it checks one or both and in which order is determined by the name service configuration of the SVM.

- For Windows to UNIX mapping

  If no mapping is found, ONTAP checks whether the lowercase Windows user name is a valid user name in the UNIX domain. If this does not work, it uses the default UNIX user provided that it is configured. If the default UNIX user is not configured and ONTAP cannot obtain a mapping this way either, mapping fails and an error is returned.

- For UNIX to Windows mapping

  If no mapping is found, ONTAP tries to find a Windows account that matches the UNIX name in the SMB domain. If this does not work, it uses the default SMB user, provided that it is configured. If the default CIFS user is not configured and ONTAP cannot obtain a mapping this way either, mapping fails and an error is returned.

Machine accounts are mapped to the specified default UNIX user by default. If no default UNIX user is specified, machine account mappings fail.

- Beginning with ONTAP 9.5, you can map machine accounts to users other than the default UNIX user.
- In ONTAP 9.4 and earlier, you cannot map machine accounts to other users.

  Even if name mappings for machine accounts are defined, the mappings are ignored.

Multidomain searches for UNIX user to Windows user name mappings

ONTAP supports multidomain searches when mapping UNIX users to Windows users. All discovered trusted domains are searched for matches to the replacement pattern until a matching result is returned. Alternatively, you can configure a list of preferred trusted domains, which is used instead of the discovered trusted domain list and is searched in order until a matching result is returned.

How domain trusts affect UNIX user to Windows user name mapping searches

To understand how multidomain user name mapping works, you must understand how domain trusts work with ONTAP. Active Directory trust relationships with the CIFS server's home domain can be a bidirectional trust or can be one of two types of unidirectional trusts, either an inbound trust or an outbound trust. The home domain is the domain to which the CIFS server on the SVM belongs.

- *Bidirectional trust*

  With bidirectional trusts, both domains trust each other. If the CIFS server's home domain has a bidirectional trust with another domain, the home domain can authenticate and authorize a user belonging to the trusted domain and vice versa.

  UNIX user to Windows user name mapping searches can be performed only on domains with bidirectional trusts between the home domain and the other domain.

- *Outbound trust*

  With an outbound trust, the home domain trusts the other domain. In this case, the home domain can authenticate and authorize a user belonging to the outbound trusted domain.

  A domain with an outbound trust with the home domain is *not* searched when performing UNIX user to Windows user name mapping searches.

- *Inbound trust*

  With an inbound trust, the other domain trusts the CIFS server's home domain. In this case, the home domain cannot authenticate or authorize a user belonging to the inbound trusted domain.

  A domain with an inbound trust with the home domain is *not* searched when performing UNIX user to Windows user name mapping searches.

How wildcards (*) are used to configure multidomain searches for name mapping

Multidomain name mapping searches are facilitated by the use of wildcards in the domain section of the Windows user name. The following table illustrates how to use wildcards in the domain part of a name mapping entry to enable multidomain searches:

Pattern	Replacement	Result
root	*\\administrator	The UNIX user "root" is mapped to the user named "administrator". All trusted domains are searched in order until the first matching user named "administrator" is found.

Pattern	Replacement	Result
*	**	Valid UNIX users are mapped to the corresponding Windows users. All trusted domains are searched in order until the first matching user with that name is found. The pattern ** is only valid for name mapping from UNIX to Windows, not the other way around.

How multidomain name searches are performed

You can choose one of two methods for determining the list of trusted domains used for multidomain name searches:

- Use the automatically discovered bidirectional trust list compiled by ONTAP
- Use the preferred trusted domain list that you compile

If a UNIX user is mapped to a Windows user with a wildcard used for the domain section of the user name, the Windows user is looked up in all the trusted domains as follows:

- If a preferred trusted-domain list is configured, the mapped Windows user is looked up in this search list only, in order.
- If a preferred list of trusted domains is not configured, then the Windows user is looked up in all the bidirectional trusted domains of the home domain.
- If there are no bidirectionally trusted domains for the home domain, the user is looked up in the home domain.

If a UNIX user is mapped to a Windows user without a domain section in the user name, the Windows user is looked up in the home domain.

Name mapping conversion rules

An ONTAP system keeps a set of conversion rules for each SVM. Each rule consists of two pieces: a *pattern* and a *replacement*. Conversions start at the beginning of the appropriate list and perform a substitution based on the first matching rule. The pattern is a UNIX-style regular expression. The replacement is a string containing escape sequences representing subexpressions from the pattern, as in the UNIX `sed` program.

Create a name mapping

You can use the `vserver name-mapping create` command to create a name mapping. You use name mappings to enable Windows users to access UNIX security style volumes and the reverse.

About this task

For each SVM, ONTAP supports up to 12,500 name mappings for each direction.

Step

1. Create a name mapping: `vserver name-mapping create -vserver vserver_name -direction {krb-unix|win-unix|unix-win} -position integer -pattern text -replacement text`

> The `-pattern` and `-replacement` statements can be formulated as regular expressions. You can also use the `-replacement` statement to explicitly deny a mapping to the user by using the null replacement string `" "` (the space character). See the `vserver name-mapping create` man page for details.

When Windows-to-UNIX mappings are created, any SMB clients that have open connections to the ONTAP system at the time the new mappings are created must log out and log back in to see the new mappings.

Examples

The following command creates a name mapping on the SVM named vs1. The mapping is a mapping from UNIX to Windows at position 1 in the priority list. The mapping maps the UNIX user johnd to the Windows user ENG\JohnDoe.

```
vs1::> vserver name-mapping create -vserver vs1 -direction unix-win
-position 1 -pattern johnd
-replacement "ENG\\JohnDoe"
```

The following command creates another name mapping on the SVM named vs1. The mapping is a mapping from Windows to UNIX at position 1 in the priority list. Here the pattern and replacement include regular expressions. The mapping maps every CIFS user in the domain ENG to users in the LDAP domain associated with the SVM.

```
vs1::> vserver name-mapping create -vserver vs1 -direction win-unix
-position 1 -pattern "ENG\\(.+)"
-replacement "\1"
```

The following command creates another name mapping on the SVM named vs1. Here the pattern includes “$” as an element in the Windows user name that must be escaped. The mapping maps the windows user ENG\john$ops to UNIX user john_ops.

```
vs1::> vserver name-mapping create -direction win-unix -position 1
-pattern ENG\\john\$ops
-replacement john_ops
```

Configure the default user

You can configure a default user to use if all other mapping attempts fail for a user, or if

you do not want to map individual users between UNIX and Windows. Alternatively, if you want authentication of non-mapped users to fail, you should not configure a default user.

About this task

For CIFS authentication, if you do not want to map each Windows user to an individual UNIX user, you can instead specify a default UNIX user.

For NFS authentication, if you do not want to map each UNIX user to an individual Windows user, you can instead specify a default Windows user.

Steps

1. Perform one of the following actions:

If you want to…	Enter the following command…
Configure the default UNIX user	`vserver cifs options modify -default -unix-user` *`user_name`*
Configure the default Windows user	`vserver nfs modify -default-win-user` *`user_name`*

Commands for managing name mappings

There are specific ONTAP commands for managing name mappings.

If you want to…	Use this command…
Create a name mapping	`vserver name-mapping create`
Insert a name mapping at a specific position	`vserver name-mapping insert`
Display name mappings	`vserver name-mapping show`
Exchange the position of two name mappings A swap is not allowed when name-mapping is configured with an ip-qualifier entry.	`vserver name-mapping swap`
Modify a name mapping	`vserver name-mapping modify`
Delete a name mapping	`vserver name-mapping delete`

If you want to…	Use this command…
Validate the correct name mapping	`vserver security file-directory show-effective-permissions -vserver vs1 -win-user-name user1 -path / -share-name sh1`

See the man page for each command for more information.

Configure multidomain name-mapping searches

Enable or disable multidomain name mapping searches

With multidomain name mapping searches, you can use a wild card (*) in the domain portion of a Windows name when configuring UNIX user to Windows user name mapping. Using a wild card (*) in the domain portion of the name enables ONTAP to search all domains that have a bidirectional trust with the domain that contains the CIFS server's computer account.

About this task

As an alternative to searching all bidirectionally trusted domains, you can configure a list of preferred trusted domains. When a list of preferred trusted domains is configured, ONTAP uses the preferred trusted domain list instead of the discovered bidirectionally trusted domains to perform multidomain name mapping searches.

- Multidomain name mapping searches are enabled by default.
- This option is available at the advanced privilege level.

Steps

1. Set the privilege level to advanced: `set -privilege advanced`
2. Perform one of the following actions:

If you want multidomain name mapping searches to be…	Enter the command…
Enabled	`vserver cifs options modify -vserver` *`vserver_name`* `-is-trusted-domain-enum -search-enabled true`
Disabled	`vserver cifs options modify -vserver` *`vserver_name`* `-is-trusted-domain-enum -search-enabled false`

3. Return to the admin privilege level: `set -privilege admin`

Related information

Available SMB server options

Reset and rediscover trusted domains

You can force the rediscovery of all the trusted domains. This can be useful when the trusted domain servers are not responding appropriately or the trust relationships have changed. Only domains with a bidirectional trust with the home domain, which is the domain containing the CIFS server's computer account, are discovered.

Step

1. Reset and rediscover trusted domains by using the `vserver cifs domain trusts rediscover` command.

   ```
   vserver cifs domain trusts rediscover -vserver vs1
   ```

Related information

Displaying information about discovered trusted domains

Display information about discovered trusted domains

You can display information about the discovered trusted domains for the CIFS server's home domain, which is the domain containing the CIFS server's computer account. This can be useful when you want to know which trusted domains are discovered and how they are ordered within the discovered trusted-domain list.

About this task

Only the domains with bidirectional trusts with the home domain are discovered. Since the home domain's domain controller (DC) returns the list of trusted domains in an order determined by the DC, the order of the domains within the list cannot be predicted. By displaying the list of trusted domains, you can determine the search order for multidomain name mapping searches.

The displayed trusted domain information is grouped by node and storage virtual machine (SVM).

Step

1. Display information about discovered trusted domains by using the `vserver cifs domain trusts show` command.

   ```
   vserver cifs domain trusts show -vserver vs1
   ```

```
   Node: node1
Vserver: vs1

Home Domain             Trusted Domain
---------------------   ----------------------------------
EXAMPLE.COM             CIFS1.EXAMPLE.COM,
                        CIFS2.EXAMPLE.COM
                        EXAMPLE.COM

   Node: node2
Vserver: vs1

Home Domain             Trusted Domain
---------------------   ----------------------------------
EXAMPLE.COM             CIFS1.EXAMPLE.COM,
                        CIFS2.EXAMPLE.COM
                        EXAMPLE.COM
```

Related information

Resetting and rediscovering trusted domains

Add, remove, or replace trusted domains in preferred trusted domain lists

You can add or remove trusted domains from the preferred trusted domain list for the SMB server or you can modify the current list. If you configure a preferred trusted domain list, this list is used instead of the discovered bidirectional trusted domains when performing multidomain name mapping searches.

About this task

- If you are adding trusted domains to an existing list, the new list is merged with the existing list with the new entries placed at the end. The trusted domains are searched in the order they appear in the trusted domain list.
- If you are removing trusted domains from the existing list and do not specify a list, the entire trusted domain list for the specified storage virtual machine (SVM) is removed.
- If you modify the existing list of trusted domains, the new list overwrites the existing list.

> You should enter only bidirectionally trusted domains in the preferred trusted domain list. Even though you can enter outbound or inbound trust domains into the preferred domain list, they are not used when performing multidomain name mapping searches. ONTAP skips the entry for the unidirectional domain and moves on to the next bidirectional trusted domain in the list.

Step

1. Perform one of the following actions:

If you want to do the following with the list of preferred trusted domains…	Use the command…
Add trusted domains to the list	`vserver cifs domain name-mapping-search add -vserver _vserver_name_ -trusted-domains FQDN, ...`
Remove trusted domains from the list	`vserver cifs domain name-mapping-search remove -vserver _vserver_name_ [-trusted-domains FQDN, ...]`
Modify the existing list	`vserver cifs domain name-mapping-search modify -vserver _vserver_name_ -trusted-domains FQDN, ...`

Examples

The following command adds two trusted domains (cifs1.example.com and cifs2.example.com) to the preferred trusted domain list used by SVM vs1:

```
cluster1::> vserver cifs domain name-mapping-search add -vserver vs1
-trusted-domains cifs1.example.com, cifs2.example.com
```

The following command removes two trusted domains from the list used by SVM vs1:

```
cluster1::> vserver cifs domain name-mapping-search remove -vserver vs1
-trusted-domains cifs1.example.com, cifs2.example.com
```

The following command modifies the trusted domain list used by SVM vs1. The new list replaces the original list:

```
cluster1::> vserver cifs domain name-mapping-search modify -vserver vs1
-trusted-domains cifs3.example.com
```

Related information

Displaying information about the preferred trusted domain list

Display information about the preferred trusted domain list

You can display information about which trusted domains are in the preferred trusted domain list and the order in which they are searched if multidomain name mapping searches are enabled. You can configure a preferred trusted domain list as an alternative to using the automatically discovered trusted domain list.

Steps

1. Perform one of the following actions:

If you want to display information about the following…	Use the command…
All preferred trusted domains in the cluster grouped by storage virtual machine (SVM)	`vserver cifs domain name-mapping-search show`
All preferred trusted domains for a specified SVM	`vserver cifs domain name-mapping-search show -vserver vserver_name`

The following command displays information about all preferred trusted domains on the cluster:

```
cluster1::> vserver cifs domain name-mapping-search show
Vserver         Trusted Domains
--------------  ---------------------------------
vs1             CIFS1.EXAMPLE.COM
```

Related information

Adding, removing, or replacing trusted domains in preferred trusted domain lists

Create and configure SMB shares

Create and configure SMB shares overview

Before users and applications can access data on the CIFS server over SMB, you must create and configure SMB shares, which is a named access point in a volume. You can customize shares by specifying share parameters and share properties. You can modify an existing share at any time.

When you create an SMB share, ONTAP creates a default ACL for the share with Full Control permissions for Everyone.

SMB shares are tied to the CIFS server on the storage virtual machine (SVM). SMB shares are deleted if either the SVM is deleted or the CIFS server with which it is associated is deleted from the SVM. If you recreate the CIFS server on the SVM, you must re-create the SMB shares.

Related information

Manage file access using SMB

SMB configuration for Microsoft Hyper-V and SQL Server

Configure character mapping for SMB file name translation on volumes

What the default administrative shares are

When you create a CIFS server on your storage virtual machine (SVM), default

administrative shares are automatically created. You should understand what those default shares are and how they are used.

ONTAP creates the following default administrative shares when you create the CIFS server:

> Beginning with ONTAP 9.8, the admin$ share is no longer created by default.

- ipc$
- admin$ (ONTAP 9.7 and earlier only)
- c$

Because shares that end with the $ character are hidden shares, the default administrative shares are not visible from My Computer, but you can view them by using Shared Folders.

How the ipc$ and admin$ default shares are used

The ipc$ and admin$ shares are used by ONTAP and cannot be used by Windows administrators to access data residing on the SVM.

- ipc$ share

  The ipc$ share is a resource that shares the named pipes that are essential for communication between programs. The ipc$ share is used during remote administration of a computer and when viewing a computer's shared resources. You cannot change the share settings, share properties, or ACLs of the ipc$ share. You also cannot rename or delete the ipc$ share.

- admin$ share (ONTAP 9.7 and earlier only)

  > Beginning with ONTAP 9.8, the admin$ share is no longer created by default.

  The admin$ share is used during remote administration of the SVM. The path of this resource is always the path to the SVM root. You cannot change the share settings, share properties, or ACLs for the admin$ share. You also cannot rename or delete the admin$ share.

How the c$ default share is used

The c$ share is an administrative share that the cluster or SVM administrator can use to access and manage the SVM root volume.

The following are characteristics of the c$ share:

- The path for this share is always the path to the SVM root volume and cannot be modified.
- The default ACL for the c$ share is Administrator / Full Control.

  This user is the BUILTIN\administrator. By default, the BUILTIN\administrator can map to the share and view, create, modify, or delete files and folders in the mapped root directory. Caution should be exercised when managing files and folders in this directory.

- You can change the c$ share's ACL.
- You can change the c$ share settings and share properties.
- You cannot delete the c$ share.

- The SVM administrator can access the rest of the SVM namespace from the mapped c$ share by crossing the namespace junctions.
- The c$ share can be accessed by using the Microsoft Management Console.

Related information

Configuring advanced NTFS file permissions using the Windows Security tab

SMB share naming requirements

You should keep the ONTAP share naming requirements in mind when creating SMB shares on your SMB server.

Share naming conventions for ONTAP are the same as for Windows and include the following requirements:

- The name of each share must be unique for the SMB server.
- Share names are not case-sensitive.
- The maximum share name length is 80 characters.
- Unicode share names are supported.
- Share names ending with the $ character are hidden shares.
- For ONTAP 9.7 and earlier, the admin$, ipc$, and c$ administrative shares are automatically created on every CIFS server and are reserved share names. Beginning with ONTAP 9.8, the admin$ share is no longer automatically created.
- You cannot use the share name ONTAP_ADMIN$ when creating a share.
- Share names containing spaces are supported:
  - You cannot use a space as the first character or as the last character in a share name.
  - You must enclose share names containing a space in quotation marks.

  Single quotation marks are considered part of the share name and cannot be used in place of quotation marks.

- The following special characters are supported when you name SMB shares:

  ! @ # $ % & ' _ - . ~ ( ) { }

- The following special characters are not supported when you name SMB shares:

  " / \ : ; | < > , ? * =

Directory case-sensitivity requirements when creating shares in a multiprotocol environment

If you create shares in an SVM where the 8.3 naming scheme is used to distinguish between directory names where there are only case differences between the names, you must use the 8.3 name in the share path to ensure that the client connects to the desired directory path.

In the following example, two directories named “testdir” and “TESTDIR” were created on a Linux client. The junction path of the volume containing the directories is `/home`. The first output is from a Linux client and the

second output is from an SMB client.

```
 ls -l
drwxrwxr-x 2 user1 group1   4096 Apr 17 11:23 testdir
drwxrwxr-x 2 user1 group1   4096 Apr 17 11:24 TESTDIR
```

```
dir

 Directory of Z:\

04/17/2015  11:23 AM    <DIR>          testdir
04/17/2015  11:24 AM    <DIR>          TESTDI~1
```

When you create a share to the second directory, you must use the 8.3 name in the share path. In this example, the share path to the first directory is `/home/testdir` and the share path to the second directory is `/home/TESTDI~1`.

Use SMB share properties

Use SMB share properties overview

You can customize the properties of SMB shares.

The available share properties are as follows:

Share properties	Description
`oplocks`	This property specifies that the share uses opportunistic locks, also known as client-side caching.
`browsable`	This property allows Windows clients to browse the share.
`showsnapshot`	This property specifies that Snapshot copies can be viewed and traversed by clients.
`changenotify`	This property specifies that the share supports Change Notify requests. For shares on an SVM, this is a default initial property.
`attributecache`	This property enables the file attribute caching on the SMB share to provide faster access of attributes. The default is to disable attribute caching. This property should be enabled only if there are clients connecting to shares over SMB 1.0. This share property is not applicable if clients are connecting to shares over SMB 2.x or SMB 3.0.

Share properties	Description
`continuously-available`	This property permits SMB clients that support it to open files in a persistent manner. Files opened this way are protected from disruptive events, such as failover and giveback.
`branchcache`	This property specifies that the share allows clients to request BranchCache hashes on the files within this share. This option is useful only if you specify “per-share” as the operating mode in the CIFS BranchCache configuration.
`access-based-enumeration`	This property specifies that *Access Based Enumeration* (ABE) is enabled on this share. ABE-filtered shared folders are visible to a user based on that individual user’s access rights, preventing the display of folders or other shared resources that the user does not have rights to access.
`namespace-caching`	This property specifies that the SMB clients connecting to this share can cache the directory enumeration results returned by the CIFS servers, which can provide better performance. By default, SMB 1 clients do not cache directory enumeration results. Because SMB 2 and SMB 3 clients cache directory enumeration results by default, specifying this share property provides performance benefits only to SMB 1 client connections.
`encrypt-data`	This property specifies that SMB encryption must be used when accessing this share. SMB clients that do not support encryption when accessing SMB data will not be able to access this share.

Add or remove share properties on an existing SMB share

You can customize an existing SMB share by adding or removing share properties. This can be useful if you want to change the share configuration to meet changing requirements in your environment.

Before you begin

The share whose properties you want to modify must exist.

About this task

Guidelines for adding share properties:

- You can add one or more share properties by using a comma-delimited list.
- Any share properties that you have previously specified remain in effect.

Newly added properties are appended to the existing list of share properties.

- If you specify a new value for share properties that are already applied to the share, the newly specified value replaces the original value.
- You cannot remove share properties by using the `vserver cifs share properties add` command.

  You can use the `vserver cifs share properties remove` command to remove share properties.

Guidelines for removing share properties:

- You can remove one or more share properties by using a comma-delimited list.
- Any share properties that you have previously specified but do not remove remain in effect.

Steps

1. Enter the appropriate command:

If you want to…	Enter the command…
Add share properties	`vserver cifs share properties add -vserver _vserver_name_ -share-name _share_name_ -share-properties _properties_,...`
Remove share properties	`vserver cifs share properties remove -vserver _vserver_name_ -share-name _share_name_ -share-properties _properties_,...`

2. Verify the share property settings: `vserver cifs share show -vserver vserver_name -share -name share_name`

Examples

The following command adds the `showsnapshot` share property to a share named “share1” on SVM vs1:

```
cluster1::> vserver cifs share properties add -vserver vs1 -share-name
share1 -share-properties showsnapshot

cluster1::> vserver cifs share show -vserver vs1
Vserver    Share   Path      Properties   Comment    ACL
---------  ------  --------  ----------   --------   -----------
vs1        share1  /share1   oplocks      -          Everyone / Full
Control
                             browsable
                             changenotify
                             showsnapshot
```

The following command removes the `browsable` share property from a share named “share2” on SVM vs1:

```
cluster1::> vserver cifs share properties remove -vserver vs1 -share-name
share2 -share-properties browsable

cluster1::> vserver cifs share show -vserver vs1
Vserver    Share   Path      Properties   Comment    ACL
---------  ------  --------  ----------   --------   -----------
vs1        share2  /share2   oplocks      -          Everyone / Full
Control
                             changenotify
```

Related information

Commands for managing SMB shares

Optimize SMB user access with the force-group share setting

When you create a share from the ONTAP command line to data with UNIX effective security, you can specify that all files created by SMB users in that share belong to the same group, known as the *force-group*, which must be a predefined group in the UNIX group database. Using a force-group makes it easier to ensure that files can be accessed by SMB users belonging to various groups.

Specifying a force-group is meaningful only if the share is in a UNIX or mixed qtree. There is no need to set a force-group for shares in an NTFS volume or qtree because access to files in these shares is determined by Windows permissions, not UNIX GIDs.

If a force-group has been specified for a share, the following becomes true of the share:

- SMB users in the force-group who access this share are temporarily changed to the GID of the force-group.

  This GID enables them to access files in this share that are not accessible normally with their primary GID or UID.

- All files in this share created by SMB users belong to the same force-group, regardless of the primary GID of the file owner.

When SMB users try to access a file created by NFS, the SMB users' primary GIDs determine access rights.

The force-group does not affect how NFS users access files in this share. A file created by NFS acquires the GID from the file owner. Determination of access permissions is based on the UID and primary GID of the NFS user who is trying to access the file.

Using a force-group makes it easier to ensure that files can be accessed by SMB users belonging to various groups. For example, if you want to create a share to store the company’s web pages and give write access to users in the Engineering and Marketing departments, you can create a share and give write access to a force-group named “webgroup1”. Because of the force-group, all files created by SMB users in this share are owned by the “webgroup1” group. In addition, users are automatically assigned the GID of the “webgroup1” group when accessing the share. As a result, all the users can write to this share without you needing to manage the access rights of the users in the Engineering and Marketing departments.

Related information

Creating an SMB share with the force-group share setting

Create an SMB share with the force-group share setting

You can create an SMB share with the force-group share setting if you want SMB users that access data on volumes or qtrees with UNIX file security to be regarded by ONTAP as belonging to the same UNIX group.

Step

1. Create the SMB share: `vserver cifs share create -vserver` *`vserver_name`* `-share-name` *`share_name`* `-path path -force-group-for-create` *`UNIX_group_name`*

   If the UNC path (`\\servername\sharename\filepath`) of the share contains more than 256 characters (excluding the initial "`\\`" in the UNC path), then the **Security** tab in the Windows Properties box is unavailable. This is a Windows client issue rather than an ONTAP issue. To avoid this issue, do not create shares with UNC paths with more than 256 characters.

   If you want to remove the force-group after the share is created, you can modify the share at any time and specify an empty string ("") as the value for the `-force-group-for-create` parameter. If you remove the force-group by modifying the share, all existing connections to this share continue to have the previously set force-group as the primary GID.

Example

The following command creates a "webpages" share that is accessible on the web in the `/corp/companyinfo` directory in which all files that SMB users create are assigned to the webgroup1 group:

```
vserver cifs share create -vserver vs1 -share-name webpages -path
/corp/companyinfo -force-group-for-create webgroup1
```

Related information

Optimize SMB user access with the force-group share setting

View information about SMB shares using the MMC

You can view information about SMB shares on your SVM and perform some management tasks using the Microsoft Management Console (MMC). Before you can view the shares, you need to connect the MMC to the SVM.

About this task

You can perform the following tasks on shares contained within SVMs using the MMC:

- View shares
- View active sessions
- View open files
- Enumerate the list of sessions, files and tree connections in the system
- Close open files in the system
- Close open sessions

- Create/manage shares

> The views displayed by the preceding capabilities are node specific and not cluster specific. Therefore, when you use the MMC to connect to the SMB server host name (that is, cifs01.domain.local), you are routed, based on how you have set up DNS, to a single LIF within your cluster.

The following functions are not supported in MMC for ONTAP:

- Creating new local users/groups
- Managing/viewing existing local users/groups
- Viewing events or performance logs
- Storage
- Services and applications

In instances where the operation is not supported, you might experience `remote procedure call failed` errors.

FAQ: Using Windows MMC with ONTAP

Steps

1. To open Computer Management MMC on any Windows server, in the **Control Panel**, select **Administrative Tools** > **Computer Management**.
2. Select **Action** > **Connect to another computer**.

   The Select Computer dialog box appears.
3. Type the name of the storage system or click **Browse** to locate the storage system.
4. Click **OK**.

   The MMC connects to the SVM.
5. In the navigation pane, click **Shared Folders** > **Shares**.

   A list of shares on the SVM is displayed in the right display pane.
6. To display the share properties for a share, double-click the share to open the **Properties** dialog box.
7. If you cannot connect to the storage system using MMC, you can add the user to the BUILTIN\Administrators group or BUILTIN\Power Users group by using one of the following commands on the storage system:

```
cifs users-and-groups local-groups add-members -vserver <vserver_name>
-group-name BUILTIN\Administrators -member-names <domainuser>

cifs users-and-groups local-groups add-members -vserver <vserver_name>
-group-name "BUILTIN\Power Users" -member-names <domainuser>
```

Commands for managing SMB shares

You use the `vserver cifs share` and `vserver cifs share properties` commands to manage SMB shares.

If you want to…	Use this command…
Create an SMB share	`vserver cifs share create`
Display SMB shares	`vserver cifs share show`
Modify an SMB share	`vserver cifs share modify`
Delete an SMB share	`vserver cifs share delete`
Add share properties to an existing share	`vserver cifs share properties add`
Remove share properties from an existing share	`vserver cifs share properties remove`
Display information about share properties	`vserver cifs share properties show`

See the man page for each command for more information.

Secure file access by using SMB share ACLs

Guidelines for managing SMB share-level ACLs

You can change share-level ACLs to give users more or less access rights to the share. You can configure share-level ACLs by using either Windows users and groups or UNIX users and groups.

After you create a share, by default, the share-level ACL gives read access to the standard group named Everyone. Read access in the ACL means that all users in the domain and all trusted domains have read-only access to the share.

You can change a share-level ACL by using the Microsoft Management Console (MMC) on a Windows client or the ONTAP command line.

The following guidelines apply when you use the MMC:

- The user and group names specified must be Windows names.
- You can specify only Windows permissions.

The following guidelines apply when you use the ONTAP command line:

- The user and group names specified can be Windows names or UNIX names.

  If a user and group type is not specified when creating or modifying ACLs, the default type is Windows

users and groups.

- You can specify only Windows permissions.

Create SMB share access control lists

Configuring share permissions by creating access control lists (ACLs) for SMB shares enables you to control the level of access to a share for users and groups.

About this task

You can configure share-level ACLs by using local or domain Windows user or group names or UNIX user or group names.

Before creating a new ACL, you should delete the default share ACL `Everyone / Full Control`, which poses a security risk.

In workgroup mode, the local domain name is the SMB server name.

Steps

1. Delete the default share ACL:`vserver cifs share access-control delete -vserver *vserver_name* -share *share_name* -user-or-group Everyone`
2. Configure the new ACL:

If you want to configure ACLs by using a…	Enter the command…
Windows user	`vserver cifs share access-control create -vserver vserver_name -share share_name -user-group-type windows -user-or-group Windows_domain_name\user_name -permission access_right`
Windows group	`vserver cifs share access-control create -vserver vserver_name -share share_name -user-group-type windows -user-or-group Windows_domain_name\group_name -permission access_right`
UNIX user	`vserver cifs share access-control create -vserver vserver_name -share share_name -user-group-type unix-user -user-or-group UNIX_user_name -permission access_right`
UNIX group	`vserver cifs share access-control create -vserver vserver_name -share share_name -user-group-type unix-group -user-or-group UNIX_group_name -permission access_right`

3. Verify that the ACL applied to the share is correct by using the `vserver cifs share access-control show` command.

Example

The following command gives `Change` permissions to the “Sales Team” Windows group for the “sales” share on the "`vs1.example.com` "SVM:

```
cluster1::> vserver cifs share access-control create -vserver
vs1.example.com -share sales -user-or-group "DOMAIN\Sales Team"
-permission Change

cluster1::> vserver cifs share access-control show -vserver
vs1.example.com
                Share       User/Group              User/Group  Access
Vserver         Name        Name                    Type
Permission
--------------- ----------- --------------------    ---------
-----------
vs1.example.com  c$          BUILTIN\Administrators  windows
Full_Control
vs1.example.com  sales       DOMAIN\Sales Team     windows     Change
```

The following command gives `Read` permission to the “engineering” UNIX group for the “eng” share on the "`vs2.example.com` "SVM:

```
cluster1::> vserver cifs share access-control create -vserver
vs2.example.com -share eng -user-group-type unix-group -user-or-group
engineering -permission Read

cluster1::> vserver cifs share access-control show -vserver
vs2.example.com
                Share       User/Group               User/Group  Access
Vserver         Name        Name                     Type
Permission
--------------- ----------- -------------------      -----------
-----------
vs2.example.com  c$          BUILTIN\Administrators   windows
Full_Control
vs2.example.com  eng         engineering              unix-group  Read
```

The following commands give `Change` permission to the local Windows group named “Tiger Team” and `Full_Control` permission to the local Windows user named “Sue Chang” for the “datavol5” share on the "`vs1` "SVM:

```
cluster1::> vserver cifs share access-control create -vserver vs1 -share
datavol5 -user-group-type windows -user-or-group "Tiger Team" -permission
Change

cluster1::> vserver cifs share access-control create -vserver vs1 -share
datavol5 -user-group-type windows -user-or-group "Sue Chang" -permission
Full_Control

cluster1::> vserver cifs share access-control show -vserver vs1
               Share       User/Group                  User/Group  Access
Vserver        Name        Name                        Type
Permission
-------------- ----------- --------------------------- -----------
-----------
vs1            c$          BUILTIN\Administrators      windows
Full_Control
vs1            datavol5    Tiger Team         windows     Change
vs1            datavol5    Sue Chang          windows     Full_Control
```

Commands for managing SMB share access control lists

You need to know the commands for managing SMB access control lists (ACLs), which includes creating, displaying, modifying, and deleting them.

If you want to…	Use this command…
Create a new ACL	`vserver cifs share access-control create`
Display ACLs	`vserver cifs share access-control show`
Modify an ACL	`vserver cifs share access-control modify`
Delete an ACL	`vserver cifs share access-control delete`

Secure file access by using file permissions

Configure advanced NTFS file permissions using the Windows Security tab

You can configure standard NTFS file permissions on files and folders by using the **Windows Security** tab in the Windows Properties window.

Before you begin

The administrator performing this task must have sufficient NTFS permissions to change permissions on the selected objects.

About this task

Configuring NTFS file permissions is done on a Windows host by adding entries to NTFS discretionary access control lists (DACLs) that are associated with an NTFS security descriptor. The security descriptor is then applied to NTFS files and directories. These tasks are automatically handled by the Windows GUI.

Steps

1. From the **Tools** menu in Windows Explorer, select **Map network drive**.
2. Complete the **Map Network Drive** dialog box:

   a. Select a **Drive** letter.

   b. In the **Folder** box, type the CIFS server name containing the share that contains the data to which you want to apply permissions and the name of the share.

      If your CIFS server name is “CIFS_SERVER” and your share is named “share1”, you should type `\\CIFS_SERVER\share1`.

      You can specify the IP address of the data interface for the CIFS server instead of the CIFS server name.

   c. Click **Finish**.

   The drive you selected is mounted and ready with the Windows Explorer window displaying files and folders contained within the share.

3. Select the file or directory for which you want to set NTFS file permissions.
4. Right-click the file or directory, and then select **Properties**.
5. Select the **Security** tab.

   The **Security** tab displays the list of users and groups for which NTFS permission are set. The **Permissions for** box displays a list of Allow and Deny permissions in effect for each user or group selected.

6. Click **Advanced**.

   The Windows Properties window displays information about existing file permissions assigned to users and groups.

7. Click **Change Permissions**.

   The Permissions window opens.

8. Perform the desired actions:

If you want to…	Do the following…
Set up advanced NTFS permissions for a new user or group	a. Click **Add**. b. In the **Enter the object name to select** box, type the name of the user or group that you want to add. c. Click **OK**.
Change advanced NTFS permissions from a user or group	a. In the **Permissions entries:** box, select the user or group whose advanced permissions you want to change. b. Click **Edit**.
Remove advanced NTFS permissions for a user or group	a. In the **Permissions entries:** box, select the user or group that you want to remove. b. Click **Remove**. c. Skip to Step 13.

If you are adding advanced NTFS permissions on a new user or group or changing NTFS advanced permissions on an existing user or group, the Permission Entry for <Object> box opens.

9. In the **Apply to** box, select how you want to apply this NTFS file permission entry.

   If you are setting up NTFS file permissions on a single file, the **Apply to** box is not active. The **Apply to** setting defaults to **This object only**.

10. In the **Permissions** box, select the **Allow** or **Deny** boxes for the advanced permissions that you want to set on this object.

    - To allow the specified access, select the **Allow** box.
    - To not allow the specified access, select the **Deny** box. You can set permissions on the following advanced rights:
    - **Full control**

      If you choose this advanced right, all other advanced rights are automatically chosen (either Allow or Deny rights).

    - **Traverse folder / execute file**
    - **List folder / read data**
    - **Read attributes**
    - **Read extended attributes**
    - **Create files / write data**
    - **Create folders / append data**
    - **Write attributes**
    - **Write extended attributes**
    - **Delete subfolders and files**

- **Delete**
- **Read permissions**
- **Change permissions**
- **Take ownership**

> If any of the advanced permission boxes are not selectable, it is because the permissions are inherited from the parent object.

11. If you want subfolders and files of this object to inherit these permissions, select the **Apply these permissions to objects and/or containers within this container only** box.
12. Click **OK**.
13. After you finish adding, removing, or editing NTFS permissions, specify the inheritance setting for this object:
    - Select the **Include inheritable permissions from this object's parent** box.

      This is the default.

    - Select the **Replace all child object permissions with inheritable permissions from this object** box.

      This setting is not present in the Permissions box if you are setting NTFS file permissions on a single file.

      > Be cautious when selecting this setting. This setting removes all existing permissions on all child objects and replaces them with this object's permission settings. You could inadvertently remove permissions that you did not want removed. It is especially important when setting permissions in a mixed security-style volume or qtree. If child objects have a UNIX effective security style, propagating NTFS permissions to those child objects results in ONTAP changing these objects from UNIX security style to NTFS security style, and all UNIX permissions on those child objects are replaced with NTFS permissions.

    - Select both boxes.
    - Select neither box.
14. Click **OK** to close the **Permissions** box.
15. Click **OK** to close the **Advanced Security settings for <Object>** box.

    For more information about how to set advanced NTFS permissions, see your Windows documentation.

Related information

Configure and apply file security on NTFS files and folders using the CLI

Displaying information about file security on NTFS security-style volumes

Displaying information about file security on mixed security-style volumes

Displaying information about file security on UNIX security-style volumes

Configure NTFS file permissions using the ONTAP CLI

You can configure NTFS file permissions on files and directories using the ONTAP CLI. This enables you to configure NTFS file permissions without needing to connect to the data using an SMB share on a Windows Client.

You can configure NTFS file permissions by adding entries to NTFS discretionary access control lists (DACLs) that are associated with an NTFS security descriptor. The security descriptor is then applied to NTFS files and directories.

You can only configure NTFS file permissions using the command line. You cannot configure NFSv4 ACLs by using the CLI.

Steps

1. Create an NTFS security descriptor.

   ```
   vserver security file-directory ntfs create -vserver svm_name -ntfs-sd
   ntfs_security_descriptor_name -owner owner_name -group primary_group_name
   -control-flags-raw raw_control_flags
   ```

2. Add DACLs to the NTFS security descriptor.

   ```
   vserver security file-directory ntfs dacl add -vserver svm_name -ntfs-sd
   ntfs_security_descriptor_name -access-type {deny|allow} -account account_name
   -rights {no-access|full-control|modify|read-and-execute|read|write} -apply-to
   {this-folder|sub-folders|files}
   ```

3. Create a file/directory security policy.

   ```
   vserver security file-directory policy create -vserver svm_name -policy-name
   policy_name
   ```

How UNIX file permissions provide access control when accessing files over SMB

A FlexVol volume can have one of three types of security style: NTFS, UNIX, or mixed. You can access data over SMB regardless of security style; however, appropriate UNIX file permissions are needed to access data with UNIX effective security.

When data is accessed over SMB, there are several access controls used when determining whether a user is authorized to perform a requested action:

- Export permissions

  Configuring export permissions for SMB access is optional.

- Share permissions
- File permissions

  The following types of file permissions might be applied to the data on which the user wants to perform an action:

  - NTFS

- UNIX NFSv4 ACLs
- UNIX mode bits

For data with NFSv4 ACLs or UNIX mode bits set, UNIX style permissions are used to determine file access rights to the data. The SVM administrator needs to set the appropriate file permission to ensure that users have the rights to perform the desired action.

Data in a mixed security-style volume might have either NTFS or UNIX effective security style. If the data has UNIX effective security style, then NFSv4 permissions or UNIX mode bits are used when determining file access rights to the data.

Secure file access by using Dynamic Access Control (DAC)

Secure file access by using Dynamic Access Control (DAC) overview

You can secure access by using Dynamic Access Control and by creating central access policies in Active Directory and applying them to files and folders on SVMs through applied Group Policy Objects (GPOs). You can configure auditing to use central access policy staging events to see the effects of changes to central access policies before you apply them.

Additions to CIFS credentials

Before Dynamic Access Control, a CIFS credential included a security principal's (the user's) identity and Windows group membership. With Dynamic Access Control, three more types of information are added to the credential—device identity, device claims, and user claims:

- Device identity

  The analog of the user's identity information, except it is the identity and group membership of the device that the user is logging in from.

- Device claims

  Assertions about a device security principal. For example, a device claim might be that it is a member of a specific OU.

- User claims

  Assertions about a user security principal. For example, a user claim might be that their AD account is a member of a specific OU.

Central access policies

Central access policies for files enable organizations to centrally deploy and manage authorization policies that include conditional expressions using user groups, user claims, device claims, and resource properties.

For example, for accessing high business impact data, a user needs to be a full time employee and only have access to the data from a managed device. Central access policies are defined in Active Directory and distributed to file servers via the GPO mechanism.

Central access policy staging with advanced auditing

Central access policies can be “staged”, in which case they are evaluated in a “what-if” manner during file access checks. The results of what would have happened if the policy was in effect and how that differs from what is currently configured are logged as an audit event. In this way, administrators can use audit event logs to study the impact of an access policy change before actually putting the policy in play. After evaluating the impact of an access policy change, the policy can be deployed via GPOs to the desired SVMs.

Related information

Supported GPOs

Applying Group Policy Objects to CIFS servers

Enabling or disabling GPO support on a CIFS server

Displaying information about GPO configurations

Displaying information about central access policies

Displaying information about central access policy rules

Configuring central access policies to secure data on CIFS servers

Displaying information about Dynamic Access Control security

SMB and NFS auditing and security tracing

Supported Dynamic Access Control functionality

If you want to use Dynamic Access Control (DAC) on your CIFS server, you need to understand how ONTAP supports Dynamic Access Control functionality in Active Directory environments.

Supported for Dynamic Access Control

ONTAP supports the following functionality when Dynamic Access Control is enabled on the CIFS server:

Functionality	Comments
Claims into the file system	Claims are simple name and value pairs that state some truth about a user. User credentials now contain claim information, and security descriptors on files can perform access checks that include claims checks. This gives administrators a finer level of control over who can access files.
Conditional expressions to file access checks	When modifying the security parameters of a file, users can now add arbitrarily complex conditional expressions to the file’s security descriptor. The conditional expression can include checks for claims.

Functionality	Comments
Central control of file access via central access policies	Central access policies are a kind of ACL stored in Active Directory that can be tagged to a file. Access to the file is only granted if the access checks of both the security descriptor on disk and the tagged central access policy allows access.This gives administrators the ability to control access to files from a central location (AD) without having to modify the security descriptor on disk.
Central access policy staging	Adds the ability to try out security changes without affecting actual file access, by “staging” a change to the central access policies, and seeing the effect of the change in an audit report.
Support for displaying information about central access policy security by using the ONTAP CLI	Extends the `vserver security file-directory show` command to display information about applied central access policies.
Security tracing that includes central access policies	Extends the `vserver security trace` command family to display results that include information about applied central access policies.

Unsupported for Dynamic Access Control

ONTAP does not support the following functionality when Dynamic Access Control is enabled on the CIFS server:

Functionality	Comments
Automatic classification of NTFS file system objects	This is an extension to the Windows File Classification Infrastructure that is not supported in ONTAP.
Advanced auditing other than central access policy staging	Only central access policy staging is supported for advanced auditing.

Considerations when using Dynamic Access Control and central access policies with CIFS servers

There are certain considerations you must keep in mind when using Dynamic Access Control (DAC) and central access policies to secure files and folders on CIFS servers.

NFS access can be denied to root if policy rule applies to domain\administrator user

Under certain circumstances, NFS access to root might be denied when central access policy security is applied to the data that the root user is attempting to access. The issue occurs when the central access policy contains a rule that is applied to the domain\administrator and the root account is mapped to the domain\administrator account.

Instead of applying a rule to the domain\administrator user, you should apply the rule to a group with administrative privileges, such as the domain\administrators group. In this way, you can map root to the domain\administrator account without root being impacted by this issue.

CIFS server's BUILTIN\Administrators group has access to resources when the applied central access policy is not found in Active Directory

It is possible that resources contained within the CIFS server have central access policies applied to them, but when the CIFS server uses the central access policy's SID to attempt to retrieve information from Active Directory, the SID does not match any existing central access policy SIDs in Active Directory. Under these circumstances, the CIFS server applies the local default recovery policy for that resource.

The local default recovery policy allows the CIFS server's BUILTIN\Administrators group access to that resource.

Enable or disable Dynamic Access Control overview

The option that enables you to use Dynamic Access Control (DAC) to secure objects on your CIFS server is disabled by default. You must enable the option if you want to use Dynamic Access Control on your CIFS server. If you later decide that you do not want to use Dynamic Access Control to secure objects stored on the CIFS server, you can disable the option.

About this task

Once Dynamic Access Control is enabled, the file system can contain ACLs with Dynamic Access Control-related entries. If Dynamic Access Control is disabled, the current Dynamic Access Control entries will be ignored, and new ones will not be allowed.

This option is available only at the advanced privilege level.

Step

1. Set the privilege level to advanced: `set -privilege advanced`
2. Perform one of the following actions:

If you want Dynamic Access Control to be…	Enter the command…
Enabled	`vserver cifs options modify -vserver` *`vserver_name`* `-is-dac-enabled true`
Disabled	`vserver cifs options modify -vserver` *`vserver_name`* `-is-dac-enabled false`

3. Return to the administrator privilege level: `set -privilege admin`

Related information

Configuring central access policies to secure data on CIFS servers

Manage ACLs that contain Dynamic Access Control ACEs when Dynamic Access Control is disabled

If you have resources that have ACLs applied with Dynamic Access Control ACEs and you disable Dynamic Access Control on the storage virtual machine (SVM), you must remove the Dynamic Access Control ACEs before you can manage the non-Dynamic Access Control ACEs on that resource.

About this task

After Dynamic Access Control is disabled, you cannot remove existing non-Dynamic Access Control ACEs or add new non-Dynamic Access Control ACEs until you have removed the existing Dynamic Access Control ACEs.

You can use whichever tool you normally use to manage ACLs to perform these steps.

Steps

1. Determine what Dynamic Access Control ACEs are applied to the resource.
2. Remove the Dynamic Access Control ACEs from the resource.
3. Add or remove non-Dynamic Access Control ACEs as desired from the resource.

Configure central access policies to secure data on CIFS servers

There are several steps that you must take to secure access to data on the CIFS server using central access policies, including enabling Dynamic Access Control (DAC) on the CIFS server, configuring central access policies in Active Directory, applying the central access policies to Active Directory containers with GPOs, and enabling GPOs on the CIFS server.

Before you begin

- The Active Directory must be configured to use central access policies.
- You must have sufficient access on the Active Directory domain controllers to create central access policies and to create and apply GPOs to the containers that contain the CIFS servers.
- You must have sufficient administrative access on the storage virtual machine (SVM) to execute the necessary commands.

About this task

Central access policies are defined and applied to group policy objects (GPOs) on Active Directory. You can consult the Microsoft TechNet Library for instructions about configuring central access policies and GPOs.

Microsoft TechNet Library

Steps

1. Enable Dynamic Access Control on the SVM if it is not already enabled by using the `vserver cifs options modify` command.

   ```
   vserver cifs options modify -vserver vs1 -is-dac-enabled true
   ```

2. Enable group policy objects (GPOs) on the CIFS server if they are not already enabled by using the `vserver cifs group-policy modify` command.

```
vserver cifs group-policy modify -vserver vs1 -status enabled
```

3. Create central access rules and central access policies on Active Directory.
4. Create a group policy object (GPO) to deploy the central access policies on Active Directory.
5. Apply the GPO to the container where the CIFS server computer account is located.
6. Manually update the GPOs applied to the CIFS server by using the `vserver cifs group-policy update` command.

   ```
   vserver cifs group-policy update -vserver vs1
   ```

7. Verify that the GPO central access policy is applied to the resources on the CIFS server by using the `vserver cifs group-policy show-applied` command.

   The following example shows that the Default Domain Policy has two central access policies that are applied to the CIFS server:

   ```
   vserver cifs group-policy show-applied
   ```

   ```
   Vserver: vs1
   ----------------------------
       GPO Name: Default Domain Policy
          Level: Domain
         Status: enabled
     Advanced Audit Settings:
         Object Access:
             Central Access Policy Staging: failure
     Registry Settings:
         Refresh Time Interval: 22
         Refresh Random Offset: 8
         Hash Publication Mode for BranchCache: per-share
         Hash Version Support for BranchCache: all-versions
     Security Settings:
         Event Audit and Event Log:
             Audit Logon Events: none
             Audit Object Access: success
             Log Retention Method: overwrite-as-needed
             Max Log Size: 16384
         File Security:
             /vol1/home
             /vol1/dir1
         Kerberos:
             Max Clock Skew: 5
             Max Ticket Age: 10
             Max Renew Age:  7
         Privilege Rights:
             Take Ownership: usr1, usr2
             Security Privilege: usr1, usr2
   ```

```
        Change Notify: usr1, usr2
    Registry Values:
        Signing Required: false
    Restrict Anonymous:
        No enumeration of SAM accounts: true
        No enumeration of SAM accounts and shares: false
        Restrict anonymous access to shares and named pipes: true
        Combined restriction for anonymous user: no-access
    Restricted Groups:
        gpr1
        gpr2
Central Access Policy Settings:
    Policies: cap1
              cap2

  GPO Name: Resultant Set of Policy
     Level: RSOP
Advanced Audit Settings:
    Object Access:
        Central Access Policy Staging: failure
Registry Settings:
    Refresh Time Interval: 22
    Refresh Random Offset: 8
    Hash Publication Mode for BranchCache: per-share
    Hash Version Support for BranchCache: all-versions
Security Settings:
    Event Audit and Event Log:
        Audit Logon Events: none
        Audit Object Access: success
        Log Retention Method: overwrite-as-needed
        Max Log Size: 16384
    File Security:
        /vol1/home
        /vol1/dir1
    Kerberos:
        Max Clock Skew: 5
        Max Ticket Age: 10
        Max Renew Age:  7
    Privilege Rights:
        Take Ownership: usr1, usr2
        Security Privilege: usr1, usr2
        Change Notify: usr1, usr2
    Registry Values:
        Signing Required: false
    Restrict Anonymous:
        No enumeration of SAM accounts: true
```

```
          No enumeration of SAM accounts and shares: false
          Restrict anonymous access to shares and named pipes: true
          Combined restriction for anonymous user: no-access
      Restricted Groups:
          gpr1
          gpr2
  Central Access Policy Settings:
      Policies: cap1
                cap2
2 entries were displayed.
```

Related information

Displaying information about GPO configurations

Displaying information about central access policies

Displaying information about central access policy rules

Enabling or disabling Dynamic Access Control

Display information about Dynamic Access Control security

You can display information about Dynamic Access Control (DAC) security on NTFS volumes and on data with NTFS effective security on mixed security-style volumes. This includes information about conditional ACEs, resource ACEs, and central access policy ACEs. You can use the results to validate your security configuration or to troubleshoot file access issues.

About this task

You must supply the name of the storage virtual machine (SVM) and the path to the data whose file or folder security information you want to display. You can display the output in summary form or as a detailed list.

Step

1. Display file and directory security settings with the desired level of detail:

If you want to display information…	Enter the following command…
In summary form	`vserver security file-directory show -vserver vserver_name -path path`
With expanded detail	`vserver security file-directory show -vserver vserver_name -path path -expand-mask true`
Where output is displayed with group and user SIDs	`vserver security file-directory show -vserver vserver_name -path path -lookup-names false`

If you want to display information…	Enter the following command…
About file and directory security for files and directories where the hexadecimal bit mask is translated to textual format	`vserver security file-directory show -vserver vserver_name -path path -textual-mask true`

Examples

The following example displays Dynamic Access Control security information about the path `/vol1` in SVM vs1:

```
cluster1::> vserver security file-directory show -vserver vs1 -path /vol1
                           Vserver: vs1
                         File Path: /vol1
                 File Inode Number: 112
                    Security Style: mixed
                   Effective Style: ntfs
                    DOS Attributes: 10
            DOS Attributes in Text: ----D---
            Expanded Dos Attribute: -
                      Unix User Id: 0
                     Unix Group Id: 1
                    Unix Mode Bits: 777
            Unix Mode Bits in Text: rwxrwxrwx
                              ACLs: NTFS Security Descriptor
                                    Control:0xbf14
                                    Owner:CIFS1\Administrator
                                    Group:CIFS1\Domain Admins
                                    SACL - ACEs
                                       ALL-Everyone-0xf01ff-OI|CI|SA|FA
                                       RESOURCE ATTRIBUTE-Everyone-0x0

("Department_MS",TS,0x10020,"Finance")
                                       POLICY ID-All resources - No Write-
0x0-OI|CI
                                    DACL - ACEs
                                       ALLOW-CIFS1\Administrator-0x1f01ff-
OI|CI
                                       ALLOW-Everyone-0x1f01ff-OI|CI
                                       ALLOW CALLBACK-DAC\user1-0x1200a9-
OI|CI

((@User.department==@Resource.Department_MS&&@Resource.Impact_MS>1000)&&@D
evice.department==@Resource.Department_MS)
```

Related information

Displaying information about GPO configurations

Displaying information about central access policies

Displaying information about central access policy rules

Revert considerations for Dynamic Access Control

You should be aware of what happens when reverting to a version of ONTAP that does not support Dynamic Access Control (DAC) and what you must do before and after reverting.

If you want to revert the cluster to a version of ONTAP that does not support Dynamic Access Control and Dynamic Access Control is enabled on one or more the storage virtual machines (SVMs), you must do the following before reverting:

- You must disable Dynamic Access Control on all SVMs that have it enabled on the cluster.
- You must modify any auditing configurations on the cluster that contain the `cap-staging` event type to use only the `file-op` event type.

You must understand and act on some important revert considerations for files and folders with Dynamic Access Control ACEs:

- If the cluster is reverted, existing Dynamic Access Control ACEs are not removed; however, they will be ignored in file access checks.
- Since Dynamic Access Control ACEs are ignored after reversion, access to files will change on files with Dynamic Access Control ACEs.

  This could allow users to access files they previously could not, or not be able to access files that they previously could.

- You should apply non-Dynamic Access Control ACEs to the affected files to restore their previous level of security.

  This can be done either before reverting or immediately after reversion completes.

> Since Dynamic Access Control ACEs are ignored after reversion, it is not required that you remove them when applying non-Dynamic Access Control ACEs to the affected files. However, if desired, you can manually remove them.

Where to find additional information about configuring and using Dynamic Access Control and central access policies

Additional resources are available to help you configure and use Dynamic Access Control and central access policies.

You can find information about how to configure Dynamic Access Control and central access policies on Active Directory in the Microsoft TechNet Library.

Microsoft TechNet: Dynamic Access Control Scenario Overview

Microsoft TechNet: Central Access Policy Scenario

The following references can help you configure the SMB server to use and support Dynamic Access Control and central access policies:

- **Using GPOs on the SMB server**

  Applying Group Policy Objects to SMB servers

- **Configuring NAS auditing on the SMB server**

  SMB and NFS auditing and security tracing

Secure SMB access using export policies

How export policies are used with SMB access

If export policies for SMB access are enabled on the SMB server, export policies are used when controlling access to SVM volumes by SMB clients. To access data, you can create an export policy that allows SMB access and then associate the policy with the volumes containing SMB shares.

An export policy has one or more rules applied to it that specifies which clients are allowed access to the data and what authentication protocols are supported for read-only and read-write access. You can configure export policies to allow access over SMB to all clients, a subnet of clients, or a specific client and to allow authentication using Kerberos authentication, NTLM authentication, or both Kerberos and NTLM authentication when determining read-only and read-write access to data.

After processing all export rules applied to the export policy, ONTAP can determine whether the client is granted access and what level of access is granted. Export rules apply to client machines, not to Windows users and groups. Export rules do not replace Windows user and group-based authentication and authorization. Export rules provide another layer of access security in addition to share and file-access permissions.

You associate exactly one export policy to each volume to configure client access to the volume. Each SVM can contain multiple export policies. This enables you to do the following for SVMs with multiple volumes:

- Assign different export policies to each volume of the SVM for individual client access control to each volume in the SVM.
- Assign the same export policy to multiple volumes of the SVM for identical client access control without having to create a new export policy for each volume.

Each SVM has at least one export policy called “default”, which contains no rules. You cannot delete this export policy, but you can rename or modify it. Each volume on the SVM by default is associated with the default export policy. If export policies for SMB access is disabled on the SVM, the “default” export policy has no effect on SMB access.

You can configure rules that provide access to both NFS and SMB hosts and associate that rule with an export policy, which can then be associated with the volume that contains data to which both NFS and SMB hosts need access. Alternatively, if there are some volumes where only SMB clients require access, you can configure an export policy with rules that only allow access using the SMB protocol and that uses only Kerberos or NTLM (or both) for authentication for read-only and write access. The export policy is then associated to the volumes where only SMB access is desired.

If export policies for SMB is enabled and a client makes an access request that is not permitted by the applicable export policy, the request fails with a permission-denied message. If a client does not match any rule in the volume's export policy, then access is denied. If an export policy is empty, then all accesses are implicitly denied. This is true even if the share and file permissions would otherwise permit access. This means that you must configure your export policy to minimally allow the following on volumes containing SMB shares:

- Allow access to all clients or the appropriate subset of clients
- Allow access over SMB
- Allow appropriate read-only and write access by using Kerberos or NTLM authentication (or both)

Learn about configuring and managing export policies.

How export rules work

Export rules are the functional elements of an export policy. Export rules match client access requests to a volume against specific parameters you configure to determine how to handle the client access requests.

An export policy must contain at least one export rule to allow access to clients. If an export policy contains more than one rule, the rules are processed in the order in which they appear in the export policy. The rule order is dictated by the rule index number. If a rule matches a client, the permissions of that rule are used and no further rules are processed. If no rules match, the client is denied access.

You can configure export rules to determine client access permissions using the following criteria:

- The file access protocol used by the client sending the request, for example, NFSv4 or SMB.
- A client identifier, for example, host name or IP address.

  The maximum size for the `-clientmatch` field is 4096 characters.

- The security type used by the client to authenticate, for example, Kerberos v5, NTLM, or AUTH_SYS.

If a rule specifies multiple criteria, the client must match all of them for the rule to apply.

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs3`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule any`
- `-rwrule any`

The client access request is sent using the NFSv3 protocol and the client has the IP address 10.1.17.37.

Even though the client access protocol matches, the IP address of the client is in a different subnet from the one specified in the export rule. Therefore, client matching fails and this rule does not apply to this client.

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule any`
- `-rwrule any`

The client access request is sent using the NFSv4 protocol and the client has the IP address 10.1.16.54.

The client access protocol matches and the IP address of the client is in the specified subnet. Therefore, client matching is successful and this rule applies to this client. The client gets read-write access regardless of its security type.

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs3`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule any`
- `-rwrule krb5,ntlm`

Client #1 has the IP address 10.1.16.207, sends an access request using the NFSv3 protocol, and authenticated with Kerberos v5.

Client #2 has the IP address 10.1.16.211, sends an access request using the NFSv3 protocol, and authenticated with AUTH_SYS.

The client access protocol and IP address matches for both clients. The read-only parameter allows read-only access to all clients regardless of the security type they authenticated with. Therefore both clients get read-only access. However, only client #1 gets read-write access because it used the approved security type Kerberos v5 to authenticate. Client #2 does not get read-write access.

Examples of export policy rules that restrict or allow access over SMB

The examples show how to create export policy rules that restrict or allow access over SMB on an SVM that has export policies for SMB access enabled.

Export policies for SMB access are disabled by default. You need to configure export policy rules that restrict or allow access over SMB only if you have enabled export policies for SMB access.

Export rule for SMB access only

The following command creates an export rule on the SVM named “vs1” that has the following configuration:

- Policy name: cifs1
- Index number: 1
- Client match: Matches only clients on the 192.168.1.0/24 network
- Protocol: Only enables SMB access
- Read-only access: To clients using NTLM or Kerberos authentication

- Read-write access: To clients using Kerberos authentication

```
cluster1::> vserver export-policy rule create -vserver vs1 -policyname
cifs1 -ruleindex 1 -protocol cifs -clientmatch 192.168.1.0/255.255.255.0
-rorule krb5,ntlm -rwrule krb5
```

Export rule for SMB and NFS access

The following command creates an export rule on the SVM named"` vs1`" that has the following configuration:

- Policy name: cifsnfs1
- Index number: 2
- Client match: Matches all clients
- Protocol: SMB and NFS access
- Read-only access: To all clients
- Read-write access: To clients using Kerberos (NFS and SMB) or NTLM authentication (SMB)
- Mapping for UNIX user ID 0 (zero): Mapped to user ID 65534 (which typically maps to the user name nobody)
- Suid and sgid access: Allows

```
cluster1::> vserver export-policy rule create -vserver vs1 -policyname
cifsnfs1 -ruleindex 2 -protocol cifs,nfs -clientmatch 0.0.0.0/0 -rorule any
-rwrule krb5,ntlm -anon 65534 -allow-suid true
```

Export rule for SMB access using NTLM only

The following command creates an export rule on the SVM named “vs1” that has the following configuration:

- Policy name: ntlm1
- Index number: 1
- Client match: Matches all clients
- Protocol: Only enables SMB access
- Read-only access: Only to clients using NTLM
- Read-write access: Only to clients using NTLM

If you configure the read-only option or the read-write option for NTLM-only access, you must use IP address-based entries in the client match option. Otherwise, you receive `access denied` errors. This is because ONTAP uses Kerberos Service Principal Names (SPN) when using a host name to check on the client’s access rights. NTLM authentication does not support SPN names.

```
cluster1::> vserver export-policy rule create -vserver vs1 -policyname
ntlm1 -ruleindex 1 -protocol cifs -clientmatch 0.0.0.0/0 -rorule ntlm
-rwrule ntlm
```

Enable or disable export policies for SMB access

You can enable or disable export policies for SMB access on storage virtual machines (SVMs). Using export policies to control SMB access to resources is optional.

Before you begin

The following are the requirements for enabling export policies for SMB:

- The client must have a “PTR” record in DNS before you create the export rules for that client.
- An additional set of “A” and “PTR” records for host names is required if the SVM provides access to NFS clients and the host name you want to use for NFS access is different from the CIFS server name.

About this task

When setting up a new CIFS server on your SVM, the use of export policies for SMB access is disabled by default. You can enable export policies for SMB access if you want to control access based on authentication protocol or on client IP addresses or host names. You can enable or disable export policies for SMB access at any time.

Steps

1. Set the privilege level to advanced: `set -privilege advanced`
2. Enable or disable export policies:
   - Enable export policies: `vserver cifs options modify -vserver` *`vserver_name`* `-is -exportpolicy-enabled true`
   - Disable export policies: `vserver cifs options modify -vserver` *`vserver_name`* `-is -exportpolicy-enabled false`
3. Return to the admin privilege level: `set -privilege admin`

Example

The following example enables the use of export policies to control SMB client access to resources on SVM vs1:

```
cluster1::> set -privilege advanced
Warning: These advanced commands are potentially dangerous; use them
only when directed to do so by technical support personnel.
Do you wish to continue? (y or n): y

cluster1::*> vserver cifs options modify -vserver vs1 -is-exportpolicy
-enabled true

cluster1::*> set -privilege admin
```

Secure file access by using Storage-Level Access Guard

Secure file access by using Storage-Level Access Guard

In addition to securing access by using native file-level and export and share security, you can configure Storage-Level Access Guard, a third layer of security applied by ONTAP at the volume level. Storage-Level Access Guard applies to access from all NAS protocols to the storage object to which it is applied.

Only NTFS access permissions are supported. For ONTAP to perform security checks on UNIX users for access to data on volumes for which Storage-Level Access Guard has been applied, the UNIX user must map to a Windows user on the SVM that owns the volume.

Storage-Level Access Guard behavior

- Storage-Level Access Guard applies to all the files or all the directories in a storage object.

  Because all files or directories in a volume are subject to Storage-Level Access Guard settings, inheritance through propagation is not required.

- You can configure Storage-Level Access Guard to apply to files only, to directories only, or to both files and directories within a volume.
  - File and directory security

    Applies to every directory and file within the storage object. This is the default setting.

  - File security

    Applies to every file within the storage object. Applying this security does not affect access to, or auditing of, directories.

  - Directory security

    Applies to every directory within the storage object. Applying this security does not affect access to, or auditing of, files.

- Storage-Level Access Guard is used to restrict permissions.

  It will never give extra access permissions.

- If you view the security settings on a file or directory from an NFS or SMB client, you do not see the Storage-Level Access Guard security.

  It’s applied at the storage object level and stored in the metadata used to determine the effective permissions.

- Storage-level security cannot be revoked from a client, even by a system (Windows or UNIX) administrator.

  It is designed to be modified by storage administrators only.

- You can apply Storage-Level Access Guard to volumes with NTFS or mixed security style.
- You can apply Storage-Level Access Guard to volumes with UNIX security style as long as the SVM containing the volume has a CIFS server configured.

- When volumes are mounted under a volume junction path and if Storage-Level Access Guard is present on that path, it will not be propagated to volumes mounted under it.
- The Storage-Level Access Guard security descriptor is replicated with SnapMirror data replication and with SVM replication.
- There is special dispensation for virus scanners.

  Exceptional access is allowed to these servers to screen files and directories, even if Storage-Level Access Guard denies access to the object.

- FPolicy notifications are not sent if access is denied because of Storage-Level Access Guard.

Order of access checks

Access to a file or directory is determined by the combined effect of the export or share permissions, the Storage-Level Access Guard permissions set on volumes, and the native file permissions applied to files and/or directories. All levels of security are evaluated to determine what the effective permissions a file or directory has. The security access checks are performed in the following order:

1. SMB share or NFS export-level permissions
2. Storage-Level Access Guard
3. NTFS file/folder access control lists (ACLs), NFSv4 ACLs, or UNIX mode bits

Use cases for using Storage-Level Access Guard

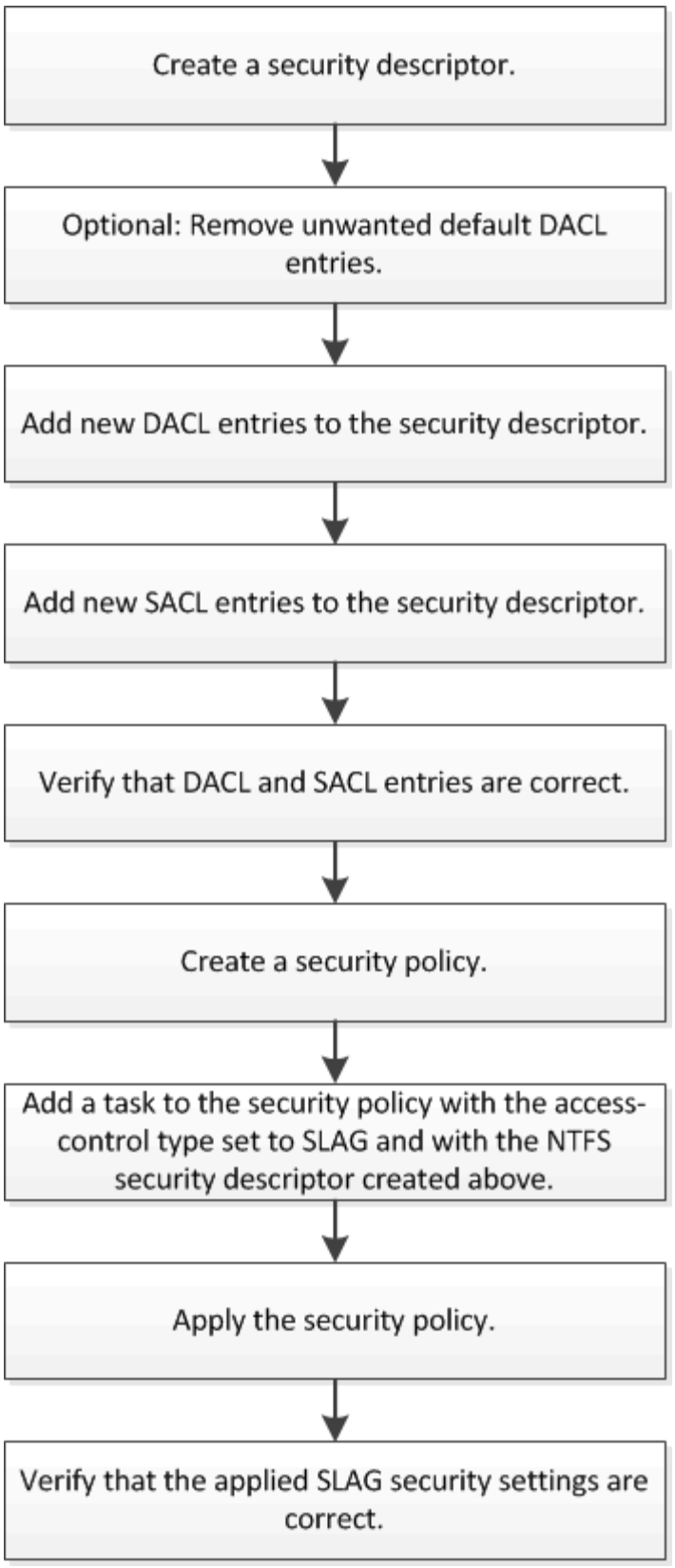
Storage-Level Access Guard provides additional security at the storage level, which is not visible from a client side; therefore, it cannot be revoked by any of the users or administrators from their desktops. There are certain use cases where the ability to control access at the storage level is beneficial.

Typical use cases for this feature include the following scenarios:

- Intellectual property protection by auditing and controlling all users' access at the storage level
- Storage for financial services companies, including banking and trading groups
- Government services with separate file storage for individual departments
- Universities protecting all student files

Workflow to configure Storage-Level Access Guard

The workflow to configure Storage-Level Access Guard (SLAG) uses the same ONTAP CLI commands that you use to configure NTFS file permissions and auditing policies. Instead of configuring file and directory access on a designated target, you configure SLAG on the designated storage virtual machine (SVM) volume.



Related information

Configuring Storage-Level Access Guard

Configure Storage-Level Access Guard

There are a number of steps you need to follow to configure Storage-Level Access Guard on a volume or qtree. Storage-Level Access Guard provides a level of access security that is set at the storage level. It provides security that applies to all accesses from all NAS protocols to the storage object to which it has been applied.

Steps

1. Create a security descriptor by using the `vserver security file-directory ntfs create` command.

   `vserver security file-directory ntfs create -vserver vs1 -ntfs-sd sd1 vserver security file-directory ntfs show -vserver vs1`

   ```
   Vserver: vs1

      NTFS Security    Owner Name
      Descriptor Name
      ------------     --------------
      sd1              -
   ```

   A security descriptor is created with the following four default DACL access control entries (ACEs):

   ```
   Vserver: vs1
     NTFS Security Descriptor Name: sd1

       Account Name     Access   Access          Apply To
                        Type     Rights
       --------------   -------  -------         -----------
       BUILTIN\Administrators
                        allow    full-control   this-folder, sub-folders,
   files
       BUILTIN\Users    allow    full-control   this-folder, sub-folders,
   files
       CREATOR OWNER    allow    full-control   this-folder, sub-folders,
   files
       NT AUTHORITY\SYSTEM
                        allow    full-control   this-folder, sub-folders,
   files
   ```

   If you do not want to use the default entries when configuring Storage-Level Access Guard, you can remove them prior to creating and adding your own ACEs to the security descriptor.

2. Remove any of the default DACL ACEs from the security descriptor that you do not want configured with Storage-Level Access Guard security:

a. Remove any unwanted DACL ACEs by using the `vserver security file-directory ntfs dacl remove` command.

   In this example, three default DACL ACEs are removed from the security descriptor: BUILTIN\Administrators, BUILTIN\Users, and CREATOR OWNER.

   ```
   vserver security file-directory ntfs dacl remove -vserver vs1 -ntfs-sd sd1
   -access-type allow -account builtin\users vserver security file-directory
   ntfs dacl remove -vserver vs1 -ntfs-sd sd1 -access-type allow -account
   builtin\administrators vserver security file-directory ntfs dacl remove
   -vserver vs1 -ntfs-sd sd1 -access-type allow -account "creator owner"
   ```

b. Verify that the DACL ACEs you do not want to use for Storage-Level Access Guard security are removed from the security descriptor by using the `vserver security file-directory ntfs dacl show` command.

   In this example, the output from the command verifies that three default DACL ACEs have been removed from the security descriptor, leaving only the NT AUTHORITY\SYSTEM default DACL ACE entry:

   ```
   vserver security file-directory ntfs dacl show -vserver vs1
   ```

   ```
   Vserver: vs1
     NTFS Security Descriptor Name: sd1

       Account Name     Access   Access          Apply To
                        Type     Rights
       --------------   -------  -------         -----------
       NT AUTHORITY\SYSTEM
                        allow    full-control   this-folder, sub-
   folders, files
   ```

3. Add one or more DACL entries to a security descriptor by using the `vserver security file-directory ntfs dacl add` command.

   In this example, two DACL ACEs are added to the security descriptor:

   ```
   vserver security file-directory ntfs dacl add -vserver vs1 -ntfs-sd sd1
   -access-type allow -account example\engineering -rights full-control -apply-to
   this-folder,sub-folders,files vserver security file-directory ntfs dacl add
   -vserver vs1 -ntfs-sd sd1 -access-type allow -account "example\Domain Users"
   -rights read -apply-to this-folder,sub-folders,files
   ```

4. Add one or more SACL entries to a security descriptor by using the `vserver security file-directory ntfs sacl add` command.

   In this example, two SACL ACEs are added to the security descriptor:

   ```
   vserver security file-directory ntfs sacl add -vserver vs1 -ntfs-sd sd1
   -access-type failure -account "example\Domain Users" -rights read -apply-to
   ```

```
this-folder,sub-folders,files vserver security file-directory ntfs sacl add
-vserver vs1 -ntfs-sd sd1 -access-type success -account example\engineering
-rights full-control -apply-to this-folder,sub-folders,files
```

5. Verify that the DACL and SACL ACEs are configured correctly by using the `vserver security file-directory ntfs dacl show` and `vserver security file-directory ntfs sacl show` commands, respectively.

   In this example, the following command displays information about DACL entries for security descriptor “sd1”:

   ```
   vserver security file-directory ntfs dacl show -vserver vs1 -ntfs-sd sd1
   ```

   ```
   Vserver: vs1
     NTFS Security Descriptor Name: sd1

       Account Name     Access   Access          Apply To
                        Type     Rights
       --------------   -------  -------         -----------
       EXAMPLE\Domain Users
                        allow    read           this-folder, sub-folders,
   files
       EXAMPLE\engineering
                        allow    full-control   this-folder, sub-folders,
   files
       NT AUTHORITY\SYSTEM
                        allow    full-control   this-folder, sub-folders,
   files
   ```

   In this example, the following command displays information about SACL entries for security descriptor “sd1”:

   ```
   vserver security file-directory ntfs sacl show -vserver vs1 -ntfs-sd sd1
   ```

   ```
   Vserver: vs1
     NTFS Security Descriptor Name: sd1

       Account Name     Access   Access          Apply To
                        Type     Rights
       --------------   -------  -------         -----------
       EXAMPLE\Domain Users
                        failure  read           this-folder, sub-folders,
   files
       EXAMPLE\engineering
                        success  full-control   this-folder, sub-folders,
   files
   ```

6. Create a security policy by using the `vserver security file-directory policy create` command.

   The following example creates a policy named “policy1”:

   ```
   vserver security file-directory policy create -vserver vs1 -policy-name
   policy1
   ```

7. Verify that the policy is correctly configured by using the `vserver security file-directory policy show` command.

   ```
   vserver security file-directory policy show
   ```

   ```
   Vserver          Policy Name
   ------------     --------------
   vs1              policy1
   ```

8. Add a task with an associated security descriptor to the security policy by using the `vserver security file-directory policy-task add` command with the `-access-control` parameter set to `slag`.

   Even though a policy can contain more than one Storage-Level Access Guard task, you cannot configure a policy to contain both file-directory and Storage-Level Access Guard tasks. A policy must contain either all Storage-Level Access Guard tasks or all file-directory tasks.

   In this example, a task is added to the policy named “policy1”, which is assigned to security descriptor “sd1”. It is assigned to the `/datavol1` path with the access control type set to “slag”.

   ```
   vserver security file-directory policy task add -vserver vs1 -policy-name
   policy1 -path /datavol1 -access-control slag -security-type ntfs -ntfs-mode
   propagate -ntfs-sd sd1
   ```

9. Verify that the task is configured correctly by using the `vserver security file-directory policy task show` command.

   ```
   vserver security file-directory policy task show -vserver vs1 -policy-name
   policy1
   ```

   ```
    Vserver: vs1
     Policy: policy1

      Index  File/Folder  Access           Security  NTFS       NTFS
   Security
             Path         Control          Type      Mode       Descriptor
   Name
      -----  -----------  ---------------  --------  ----------
   ---------------
      1      /datavol1    slag             ntfs      propagate  sd1
   ```

10. Apply the Storage-Level Access Guard security policy by using the `vserver security file-directory apply` command.

    ```
    vserver security file-directory apply -vserver vs1 -policy-name policy1
    ```

    The job to apply the security policy is scheduled.

11. Verify that the applied Storage-Level Access Guard security settings are correct by using the `vserver security file-directory show` command.

    In this example, the output from the command shows that Storage-Level Access Guard security has been applied to the NTFS volume `/datavol1`. Even though the default DACL allowing Full Control to Everyone remains, Storage-Level Access Guard security restricts (and audits) access to the groups defined in the Storage-Level Access Guard settings.

    ```
    vserver security file-directory show -vserver vs1 -path /datavol1
    ```

```
                Vserver: vs1
              File Path: /datavol1
      File Inode Number: 77
         Security Style: ntfs
        Effective Style: ntfs
         DOS Attributes: 10
 DOS Attributes in Text: ----D---
Expanded Dos Attributes: -
           Unix User Id: 0
          Unix Group Id: 0
         Unix Mode Bits: 777
 Unix Mode Bits in Text: rwxrwxrwx
                   ACLs: NTFS Security Descriptor
                         Control:0x8004
                         Owner:BUILTIN\Administrators
                         Group:BUILTIN\Administrators
                         DACL - ACEs
                           ALLOW-Everyone-0x1f01ff
                           ALLOW-Everyone-0x10000000-OI|CI|IO


                         Storage-Level Access Guard security
                         SACL (Applies to Directories):
                           AUDIT-EXAMPLE\Domain Users-0x120089-FA
                           AUDIT-EXAMPLE\engineering-0x1f01ff-SA
                         DACL (Applies to Directories):
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-EXAMPLE\engineering-0x1f01ff
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff
                         SACL (Applies to Files):
                           AUDIT-EXAMPLE\Domain Users-0x120089-FA
                           AUDIT-EXAMPLE\engineering-0x1f01ff-SA
                         DACL (Applies to Files):
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-EXAMPLE\engineering-0x1f01ff
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff
```

Related information

Managing NTFS file security, NTFS audit policies, and Storage-Level Access Guard on SVMs using the CLI

Workflow to configure Storage-Level Access Guard

Displaying information about Storage-Level Access Guard

Removing Storage-Level Access Guard

Effective SLAG matrix

You can configure SLAG on a volume or a qtree or both. The SLAG matrix defines on which volume or qtree is the SLAG configuration applicable under various scenarios listed in the table.

	Volume SLAG in an AFS	Volume SLAG in a Snapshot copy	Qtree SLAG in an AFS	Qtree SLAG in a Snapshot copy
Volume access in an Access File System (AFS)	YES	NO	N/A	N/A
Volume access in a Snapshot copy	YES	NO	N/A	N/A
Qtree access in an AFS (when SLAG is present in the qtree)	NO	NO	YES	NO
Qtree access in an AFS (when SLAG is not present in qtree)	YES	NO	NO	NO
Qtree access in Snapshot copy (when SLAG is present in the qtree AFS)	NO	NO	YES	NO
Qtree access in Snapshot copy (when SLAG is not present in the qtree AFS)	YES	NO	NO	NO

Display information about Storage-Level Access Guard

Storage-Level Access Guard is a third layer of security applied on a volume or qtree. Storage-Level Access Guard settings cannot be viewed by using the Windows Properties window. You must use the ONTAP CLI to view information about Storage-Level Access Guard security, which you can use to validate your configuration or to troubleshoot file access issues.

About this task

You must supply the name of the storage virtual machine (SVM) and the path to the volume or qtree whose Storage-Level Access Guard security information you want to display. You can display the output in summary form or as a detailed list.

Step

1. Display Storage-Level Access Guard security settings with the desired level of detail:

If you want to display information…	Enter the following command…
In summary form	`vserver security file-directory show -vserver vserver_name -path path`
With expanded detail	`vserver security file-directory show -vserver vserver_name -path path -expand-mask true`

Examples

The following example displays Storage-Level Access Guard security information for the NTFS security-style volume with the path `/datavol1` in SVM vs1:

```
cluster::> vserver security file-directory show -vserver vs1 -path
/datavol1

                Vserver: vs1
              File Path: /datavol1
      File Inode Number: 77
         Security Style: ntfs
        Effective Style: ntfs
         DOS Attributes: 10
 DOS Attributes in Text: ----D---
Expanded Dos Attributes: -
           Unix User Id: 0
          Unix Group Id: 0
         Unix Mode Bits: 777
 Unix Mode Bits in Text: rwxrwxrwx
                   ACLs: NTFS Security Descriptor
                         Control:0x8004
                         Owner:BUILTIN\Administrators
                         Group:BUILTIN\Administrators
                         DACL - ACEs
                           ALLOW-Everyone-0x1f01ff
                           ALLOW-Everyone-0x10000000-OI|CI|IO


                         Storage-Level Access Guard security
                         SACL (Applies to Directories):
                           AUDIT-EXAMPLE\Domain Users-0x120089-FA
                           AUDIT-EXAMPLE\engineering-0x1f01ff-SA
                         DACL (Applies to Directories):
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-EXAMPLE\engineering-0x1f01ff
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff
                         SACL (Applies to Files):
                           AUDIT-EXAMPLE\Domain Users-0x120089-FA
                           AUDIT-EXAMPLE\engineering-0x1f01ff-SA
                         DACL (Applies to Files):
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-EXAMPLE\engineering-0x1f01ff
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff
```

The following example displays the Storage-Level Access Guard information about the mixed security-style volume at the path `/datavol5` in SVM vs1. The top level of this volume has UNIX effective security. The volume has Storage-Level Access Guard security.

```
cluster1::> vserver security file-directory show -vserver vs1 -path
/datavol5
                Vserver: vs1
              File Path: /datavol5
      File Inode Number: 3374
         Security Style: mixed
        Effective Style: unix
         DOS Attributes: 10
 DOS Attributes in Text: ----D---
Expanded Dos Attributes: -
           Unix User Id: 0
          Unix Group Id: 0
         Unix Mode Bits: 755
 Unix Mode Bits in Text: rwxr-xr-x
                   ACLs: Storage-Level Access Guard security
                         SACL (Applies to Directories):
                           AUDIT-EXAMPLE\Domain Users-0x120089-FA
                           AUDIT-EXAMPLE\engineering-0x1f01ff-SA
                         DACL (Applies to Directories):
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-EXAMPLE\engineering-0x1f01ff
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff
                         SACL (Applies to Files):
                           AUDIT-EXAMPLE\Domain Users-0x120089-FA
                           AUDIT-EXAMPLE\engineering-0x1f01ff-SA
                         DACL (Applies to Files):
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-EXAMPLE\engineering-0x1f01ff
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff
```

Remove Storage-Level Access Guard

You can remove Storage-Level Access Guard on a volume or qtree if you no longer want set access security at the storage level. Removing Storage-Level Access Guard does not modify or remove regular NTFS file and directory security.

Steps

1. Verify that the volume or qtree has Storage-Level Access Guard configured by using the `vserver security file-directory show` command.

   ```
   vserver security file-directory show -vserver vs1 -path /datavol2
   ```

```
                Vserver: vs1
              File Path: /datavol2
      File Inode Number: 99
         Security Style: ntfs
        Effective Style: ntfs
         DOS Attributes: 10
 DOS Attributes in Text: ----D---
Expanded Dos Attributes: -
           Unix User Id: 0
          Unix Group Id: 0
         Unix Mode Bits: 777
 Unix Mode Bits in Text: rwxrwxrwx
                   ACLs: NTFS Security Descriptor
                         Control:0xbf14
                         Owner:BUILTIN\Administrators
                         Group:BUILTIN\Administrators
                         SACL - ACEs
                           AUDIT-EXAMPLE\Domain Users-0xf01ff-OI|CI|FA
                         DACL - ACEs
                           ALLOW-EXAMPLE\Domain Admins-0x1f01ff-OI|CI
                           ALLOW-EXAMPLE\Domain Users-0x1301bf-OI|CI

                         Storage-Level Access Guard security
                         DACL (Applies to Directories):
                           ALLOW-BUILTIN\Administrators-0x1f01ff
                           ALLOW-CREATOR OWNER-0x1f01ff
                           ALLOW-EXAMPLE\Domain Admins-0x1f01ff
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff
                         DACL (Applies to Files):
                           ALLOW-BUILTIN\Administrators-0x1f01ff
                           ALLOW-CREATOR OWNER-0x1f01ff
                           ALLOW-EXAMPLE\Domain Admins-0x1f01ff
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff
```

2. Remove Storage-Level Access Guard by using the `vserver security file-directory remove-slag` command.

   ```
   vserver security file-directory remove-slag -vserver vs1 -path /datavol2
   ```

3. Verify that Storage-Level Access Guard has been removed from the volume or qtree by using the `vserver security file-directory show` command.

   ```
   vserver security file-directory show -vserver vs1 -path /datavol2
   ```

```
               Vserver: vs1
             File Path: /datavol2
      File Inode Number: 99
         Security Style: ntfs
        Effective Style: ntfs
         DOS Attributes: 10
 DOS Attributes in Text: ----D---
Expanded Dos Attributes: -
           Unix User Id: 0
          Unix Group Id: 0
         Unix Mode Bits: 777
 Unix Mode Bits in Text: rwxrwxrwx
                   ACLs: NTFS Security Descriptor
                         Control:0xbf14
                         Owner:BUILTIN\Administrators
                         Group:BUILTIN\Administrators
                         SACL - ACEs
                           AUDIT-EXAMPLE\Domain Users-0xf01ff-OI|CI|FA
                         DACL - ACEs
                           ALLOW-EXAMPLE\Domain Admins-0x1f01ff-OI|CI
                           ALLOW-EXAMPLE\Domain Users-0x1301bf-OI|CI
```

Copyright information

Trademark information