Manage file access using SMB

ONTAP 9

NetApp
July 12, 2023

Table of Contents

Manage file access using SMB

Use local users and groups for authentication and authorization

How ONTAP uses local users and groups

Local users and groups concepts

You should know what local users and groups are, and some basic information about them, before determining whether to configure and use local users and groups in your environment.

- **Local user**

  A user account with a unique security identifier (SID) that has visibility only on the storage virtual machine (SVM) on which it is created. Local user accounts have a set of attributes, including user name and SID. A local user account authenticates locally on the CIFS server using NTLM authentication.

  User accounts have several uses:

  - Used to grant *User Rights Management* privileges to a user.
  - Used to control share-level and file-level access to file and folder resources that the SVM owns.

- **Local group**

  A group with a unique SID has visibility only on the SVM on which it is created. Groups contain a set of members. Members can be local users, domain users, domain groups, and domain machine accounts. Groups can be created, modified, or deleted.

  Groups have several uses:

  - Used to grant *User Rights Management* privileges to its members.
  - Used to control share-level and file-level access to file and folder resources that the SVM owns.

- **Local domain**

  A domain that has local scope, which is bounded by the SVM. The local domain's name is the CIFS server name. Local users and groups are contained within the local domain.

- **Security identifier (SID)**

  A SID is a variable-length numeric value that identifies Windows-style security principals. For example, a typical SID takes the following form: S-1-5-21-3139654847-1303905135-2517279418-123456.

- **NTLM authentication**

  A Microsoft Windows security method used to authenticate users on a CIFS server.

- **Cluster replicated database (RDB)**

  A replicated database with an instance on each node in a cluster. Local user and group objects are stored

in the RDB.

Reasons for creating local users and local groups

There are several reasons for creating local users and local groups on your storage virtual machine (SVM). For example, you can access an SMB server by using a local user account if the domain controllers (DCs) are unavailable, you might want to use local groups to assign privileges, or your SMB server is in a workgroup.

You can create one or more local user accounts for the following reasons:

- Your SMB server is in a workgroup, and domain users are not available.

  Local users are required in workgroup configurations.

- You want the ability to authenticate and log in to the SMB server if the domain controllers are unavailable.

  Local users can authenticate with the SMB server by using NTLM authentication when the domain controller is down, or when network problems prevent your SMB server from contacting the domain controller.

- You want to assign *User Rights Management* privileges to a local user.

  User Rights Management is the ability for an SMB server administrator to control what rights the users and groups have on the SVM. You can assign privileges to a user by assigning the privileges to the user's account, or by making the user a member of a local group that has those privileges.

You can create one or more local groups for the following reasons:

- Your SMB server is in a workgroup, and domain groups are not available.

  Local groups are not required in workgroup configurations, but they can be useful for managing access privileges for local workgroup users.

- You want to control access to file and folder resources by using local groups for share and file-access control.
- You want to create local groups with customized *User Rights Management* privileges.

  Some built-in user groups have predefined privileges. To assign a customized set of privileges, you can create a local group and assign the necessary privileges to that group. You can then add local users, domain users, and domain groups to the local group.

Related information

How local user authentication works

List of supported privileges

How local user authentication works

Before a local user can access data on a CIFS server, the user must create an authenticated session.

Because SMB is session-based, the identity of the user can be determined just once, when the session is first set up. The CIFS server uses NTLM-based authentication when authenticating local users. Both NTLMv1 and NTLMv2 are supported.

ONTAP uses local authentication under three use cases. Each use case depends on whether the domain portion of the user name (with the DOMAIN\user format) matches the CIFS server's local domain name (the CIFS server name):

- The domain portion matches

  Users who provide local user credentials when requesting access to data are authenticated locally on the CIFS server.

- The domain portion does not match

  ONTAP attempts to use NTLM authentication with a domain controller in the domain to which the CIFS server belongs. If authentication succeeds, the login is complete. If it does not succeed, what happens next depends on why authentication did not succeed.

  For example, if the user exists in Active Directory but the password is invalid or expired, ONTAP does not attempt to use the corresponding local user account on the CIFS server. Instead, authentication fails. There are other cases where ONTAP uses the corresponding local account on the CIFS server, if it exists, for authentication—even though the NetBIOS domain names do not match. For example, if a matching domain account exists but it is disabled, ONTAP uses the corresponding local account on the CIFS server for authentication.

- The domain portion is not specified

  ONTAP first attempts authentication as a local user. If authentication as a local user fails, then ONTAP authenticates the user with a domain controller in the domain to which the CIFS server belongs.

After local or domain user authentication is completed successfully, ONTAP constructs a complete user access token, which takes into account local group membership and privileges.

For more information about NTLM authentication for local users, see the Microsoft Windows documentation.

Related information

Enabling or disabling local user authentication

How user access tokens are constructed

When a user maps a share, an authenticated SMB session is established and a user access token is constructed that contains information about the user, the user's group membership and cumulative privileges, and the mapped UNIX user.

Unless the functionality is disabled, local user and group information is also added to the user access token. The way access tokens are constructed depends on whether the login is for a local user or an Active Directory domain user:

- Local user login

  Although local users can be members of different local groups, local groups cannot be members of other local groups. The local user access token is composed of a union of all privileges assigned to groups to which a particular local user is a member.

- Domain user login

  When a domain user logs in, ONTAP obtains a user access token that contains the user SID and SIDs for all the domain groups to which the user is a member. ONTAP uses the union of the domain user access token with the access token provided by local memberships of the user's domain groups (if any), as well as any direct privileges assigned to the domain user or any of its domain group memberships.

For both local and domain user login, the Primary Group RID is also set for the user access token. The default RID is `Domain Users` (RID 513). You cannot change the default.

The Windows-to-UNIX and UNIX-to-Windows name mapping process follows the same rules for both local and domain accounts.

> There is no implied, automatic mapping from a UNIX user to a local account. If this is required, an explicit mapping rule must be specified using the existing name mapping commands.

Guidelines for using SnapMirror on SVMs that contain local groups

You should be aware of the guidelines when you configure SnapMirror on volumes owned by SVMs that contain local groups.

You cannot use local groups in ACEs applied to files, directories, or shares that are replicated by SnapMirror to another SVM. If you use the SnapMirror feature to create a DR mirror to a volume on another SVM and the volume has an ACE for a local group, the ACE is not valid on the mirror. If data is replicated to a different SVM, the data is effectively crossing into a different local domain. The permissions granted to local users and groups are valid only within the scope of the SVM on which they were originally created.

What happens to local users and groups when deleting CIFS servers

The default set of local users and groups is created when a CIFS server is created, and they are associated with the storage virtual machine (SVM) hosting the CIFS server. SVM administrators can create local users and groups at any time. You need to be aware of what happens to local users and groups when you delete the CIFS server.

Local users and groups are associated with SVMs; therefore, they are not deleted when CIFS servers are deleted due to security considerations. Although local users and groups are not deleted when the CIFS server is deleted, they are hidden. You cannot view or manage local users and groups until you re-create a CIFS server on the SVM.

> The CIFS server administrative status does not affect visibility of local users or groups.

How you can use Microsoft Management Console with local users and groups

You can view information about local users and groups from the Microsoft Management Console. With this release of ONTAP, you cannot perform other management tasks for local users and groups from the Microsoft Management Console.

Guidelines for reverting

If you plan to revert the cluster to an ONTAP release that does not support local users

and groups and local users and groups are being used to manage file access or user rights, you must be aware of certain considerations.

- Due to security reasons, information about configured local users, groups, and privileges are not deleted when ONTAP is reverted to a version that does not support local users and groups functionality.
- Upon a revert to a prior major version of ONTAP, ONTAP does not use local users and groups during authentication and credential creation.
- Local users and groups are not removed from file and folder ACLs.
- File access requests that depend on access being granted because of permissions granted to local users or groups are denied.

  To allow access, you must reconfigure file permissions to allow access based on domain objects instead of local user and group objects.

What local privileges are

List of supported privileges

ONTAP has a predefined set of supported privileges. Certain predefined local groups have some of these privileges added to them by default. You can also add or remove privileges from the predefined groups or create new local users or groups and add privileges to the groups that you created or to existing domain users and groups.

The following table lists the supported privileges on the storage virtual machine (SVM) and provides a list of BUILTIN groups with assigned privileges:

Privilege name	Default security setting	Description
`SeTcbPrivilege`	None	Act as part of the operating system
`SeBackupPrivilege`	`BUILTIN\Administrators`, `BUILTIN\Backup Operators`	Back up files and directories, overriding any ACLs
`SeRestorePrivilege`	`BUILTIN\Administrators`, `BUILTIN\Backup Operators`	Restore files and directories, overriding any ACLs Set any valid user or group SID as the file owner
`SeTakeOwnershipPrivilege`	`BUILTIN\Administrators`	Take ownership of files or other objects
`SeSecurityPrivilege`	`BUILTIN\Administrators`	Manage auditingThis includes viewing, dumping, and clearing the security log.

Privilege name	Default security setting	Description
SeChangeNotifyPrivilege	BUILTIN\Administrators, BUILTIN\Backup Operators, BUILTIN\Power Users, BUILTIN\Users, Everyone	Bypass traverse checkingUsers with this privilege are not required to have traverse (x) permissions to traverse folders, symlinks, or junctions.

Related information

- Assign local privileges
- Configuring bypass traverse checking

Assign privileges

You can assign privileges directly to local users or domain users. Alternatively, you can assign users to local groups whose assigned privileges match the capabilities that you want those users to have.

- You can assign a set of privileges to a group that you create.

  You then add a user to the group that has the privileges that you want that user to have.

- You can also assign local users and domain users to predefined groups whose default privileges match the privileges that you want to grant to those users.

Related information

- Adding privileges to local or domain users or groups
- Removing privileges from local or domain users or groups
- Resetting privileges for local or domain users and groups
- Configuring bypass traverse checking

Guidelines for using BUILTIN groups and the local administrator account

There are certain guidelines you should keep in mind when you use BUILTIN groups and the local administrator account. For example, you can rename the local administrator account, but you cannot delete this account.

- The Administrator account can be renamed but cannot be deleted.
- The Administrator account cannot be removed from the BUILTIN\Administrators group.
- BUILTIN groups can be renamed but cannot be deleted.

  After the BUILTIN group is renamed, another local object can be created with the well-known name; however, the object is assigned a new RID.

- There is no local Guest account.

Related information

Predefined BUILTIN groups and default privileges

Requirements for local user passwords

By default, local user passwords must meet complexity requirements. The password complexity requirements are similar to the requirements defined in the Microsoft Windows *Local security policy*.

The password must meet the following criteria:

- Must be at least six characters in length
- Must not contain the user account name
- Must contain characters from at least three of the following four categories:
  - English uppercase characters (A through Z)
  - English lowercase characters (a through z)
  - Base 10 digits (0 through 9)
  - Special characters:

    ~ ! @ # $ % ^ & * _ - + = ` \ | ( ) [ ] : ; " ' < > , . ? /

Related information

Enabling or disabling required password complexity for local SMB users

Displaying information about CIFS server security settings

Changing local user account passwords

Predefined BUILTIN groups and default privileges

You can assign membership of a local user or domain user to a predefined set of BUILTIN groups provided by ONTAP. Predefined groups have predefined privileges assigned.

The following table describes the predefined groups:

Predefined BUILTIN group	Default privileges
`BUILTIN\Administrators`RID 544 When first created, the local `Administrator` account, with a RID of 500, is automatically made a member of this group. When the storage virtual machine (SVM) is joined to a domain, the `domain\Domain Admins` group is added to the group. If the SVM leaves the domain, the `domain\Domain Admins` group is removed from the group.	• `SeBackupPrivilege` • `SeRestorePrivilege` • `SeSecurityPrivilege` • `SeTakeOwnershipPrivilege` • `SeChangeNotifyPrivilege`

Predefined BUILTIN group	Default privileges
`BUILTIN\Power Users`RID 547 When first created, this group does not have any members. Members of this group have the following characteristics: • Can create and manage local users and groups. • Cannot add themselves or any other object to the `BUILTIN\Administrators` group.	`SeChangeNotifyPrivilege`
`BUILTIN\Backup Operators`RID 551 When first created, this group does not have any members. Members of this group can override read and write permissions on files or folders if they are opened with backup intent.	• `SeBackupPrivilege` • `SeRestorePrivilege` • `SeChangeNotifyPrivilege`
`BUILTIN\Users`RID 545 When first created, this group does not have any members (besides the implied `Authenticated Users` special group). When the SVM is joined to a domain, the `domain\Domain Users` group is added to this group. If the SVM leaves the domain, the `domain\Domain Users` group is removed from this group.	`SeChangeNotifyPrivilege`
`Everyone`SID S-1-1-0 This group includes all users, including guests (but not anonymous users). This is an implied group with an implied membership.	`SeChangeNotifyPrivilege`

Related information

Guidelines for using BUILTIN groups and the local administrator account

List of supported privileges

Configuring bypass traverse checking

Enable or disable local users and groups functionality

Enable or disable local users and groups functionality overview

Before you can use local users and groups for access control of NTFS security-style data, local user and group functionality must be enabled. Additionally, if you want to use local users for SMB authentication, the local user authentication functionality must be enabled.

Local users and groups functionality and local user authentication are enabled by default. If they are not enabled, you must enable them before you can configure and use local users and groups. You can disable local users and groups functionality at any time.

In addition to explicitly disabling local user and group functionality, ONTAP disables local user and group functionality if any node in the cluster is reverted to an ONTAP release that does not support the functionality. Local user and group functionality is not enabled until all nodes in the cluster are running a version of ONTAP that supports it.

Related information

Modify local user accounts

Modify local groups

Add privileges to local or domain users or groups

Enable or disable local users and groups

You can enable or disable local users and groups for SMB access on storage virtual machines (SVMs). Local users and groups functionality is enabled by default.

About this task

You can use local users and groups when configuring SMB share and NTFS file permissions and can optionally use local users for authentication when creating an SMB connection. To use local users for authentication, you must also enable the local users and groups authentication option.

Steps

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

If you want local users and groups to be…	Enter the command…
Enabled	`vserver cifs options modify -vserver` *`vserver_name`* `-is-local-users-and -groups-enabled true`
Disabled	`vserver cifs options modify -vserver` *`vserver_name`* `-is-local-users-and -groups-enabled false`

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

Example

The following example enables local users and groups functionality 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-local-users-and
-groups-enabled true

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

Related information

Enable or disable local user authentication

Enable or disable local user accounts

Enable or disable local user authentication

You can enable or disable local user authentication for SMB access on storage virtual machines (SVMs). The default is to allow local user authentication, which is useful when the SVM cannot contact a domain controller or if you choose not to use domain-level access controls.

Before you begin

Local users and groups functionality must be enabled on the CIFS server.

About this task

You can enable or disable local user authentication at any time. If you want to use local users for authentication when creating an SMB connection, you must also enable the CIFS server's local users and groups option.

Steps

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

If you want local authentication to be…	Enter the command…
Enabled	`vserver cifs options modify -vserver` *`vserver_name`* `-is-local-auth-enabled true`
Disabled	`vserver cifs options modify -vserver` *`vserver_name`* `-is-local-auth-enabled false`

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

Example

The following example enables local user authentication 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-local-auth
-enabled true

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

Related information

How local user authentication works

Enabling or disabling local users and groups

Manage local user accounts

Modify local user accounts

You can modify a local user account if you want to change an existing user's full name or description, and if you want to enable or disable the user account. You can also rename a local user account if the user's name is compromised or if a name change is needed for administrative purposes.

If you want to…	Enter the command…
Modify the local user's full name	`vserver cifs users-and-groups local-user modify -vserver` *`vserver_name`* `-user -name` *`user_name`* `-full-name text` If the full name contains a space, then it must be enclosed within double quotation marks.
Modify the local user's description	`vserver cifs users-and-groups local-user modify -vserver` *`vserver_name`* `-user -name` *`user_name`* `-description text` If the description contains a space, then it must be enclosed within double quotation marks.
Enable or disable the local user account	`vserver cifs users-and-groups local-user modify -vserver` *`vserver_name`* `-user -name` *`user_name`* `-is-account-disabled {true|false}`

If you want to…	Enter the command…
Rename the local user account	`vserver cifs users-and-groups local-user rename -vserver` *vserver_name* `-user-name` *user_name* `-new-user-name` *new_user_name* When renaming a local user, the new user name must remain associated with the same CIFS server as the old user name.

Example

The following example renames the local user “CIFS_SERVER\sue” to “CIFS_SERVER\sue_new” on storage virtual machine (SVM, formerly known as Vserver) vs1:

```
cluster1::> vserver cifs users-and-groups local-user rename -user-name
CIFS_SERVER\sue -new-user-name CIFS_SERVER\sue_new -vserver vs1
```

Enable or disable local user accounts

You enable a local user account if you want the user to be able to access data contained in the storage virtual machine (SVM) over an SMB connection. You can also disable a local user account if you do not want that user to access SVM data over SMB.

About this task

You enable a local user by modifying the user account.

Step

1. Perform the appropriate action:

If you want to…	Enter the command…
Enable the user account	`vserver cifs users-and-groups local-user modify -vserver` *vserver_name* `-user-name` *user_name* `-is-account -disabled false`
Disable the user account	`vserver cifs users-and-groups local-user modify -vserver` *vserver_name* `-user-name` *user_name* `-is-account -disabled true`

Change local user account passwords

You can change a local user’s account password. This can be useful if the user’s password is compromised or if the user has forgotten the password.

Step

1. Change the password by performing the appropriate action: `vserver cifs users-and-groups`

```
local-user set-password -vserver vserver_name -user-name user_name
```

Example

The following example sets the password for the local user “CIFS_SERVER\sue” associated with storage virtual machine (SVM, formerly known as Vserver) vs1:

```
cluster1::> vserver cifs users-and-groups local-user set-password -user
-name CIFS_SERVER\sue -vserver vs1

Enter the new password:
Confirm the new password:
```

Related information

Enabling or disabling required password complexity for local SMB users

Displaying information about CIFS server security settings

Display information about local users

You can display a list of all local users in a summary form. If you want to determine which account settings are configured for a specific user, you can display detailed account information for that user as well as the account information for multiple users. This information can help you determine if you need to modify a user’s settings, and also to troubleshoot authentication or file access issues.

About this task

Information about a user’s password is never displayed.

Step

1. Perform one of the following actions:

If you want to…	Enter the command…
Display information about all users on the storage virtual machine (SVM)	`vserver cifs users-and-groups local-user show -vserver vserver_name`
Display detailed account information for a user	`vserver cifs users-and-groups local-user show -instance -vserver vserver_name -user-name user_name`

   There are other optional parameters that you can choose when you run the command. See the man page for more information.

Example

The following example displays information about all local users on SVM vs1:

```
cluster1::> vserver cifs users-and-groups local-user show -vserver vs1
Vserver  User Name                    Full Name     Description
-------- -------------------------- ------------- -------------
vs1      CIFS_SERVER\Administrator   James Smith   Built-in administrator
account
vs1      CIFS_SERVER\sue             Sue   Jones
```

Display information about group memberships for local users

You can display information about which local groups that a local user belongs to. You can use this information to determine what access the user should have to files and folders. This information can be useful in determining what access rights the user should have to files and folders or when troubleshooting file access issues.

About this task

You can customize the command to display only the information that you want to see.

Step

1. Perform one of the following actions:

If you want to…	Enter the command…
Display local user membership information for a specified local user	`vserver cifs users-and-groups local-user show-membership -user-name` *user_name*
Display local user membership information for the local group of which this local user is a member	`vserver cifs users-and-groups local-user show-membership -membership` *group_name*
Display user membership information for local users that are associated with a specified storage virtual machine (SVM)	`vserver cifs users-and-groups local-user show-membership -vserver` *vserver_name*
Display detailed information for all local users on a specified SVM	`vserver cifs users-and-groups local-user show-membership -instance -vserver` *vserver_name*

Example

The following example displays the membership information for all local users on SVM vs1; user “CIFS_SERVER\Administrator” is a member of the “BUILTIN\Administrators” group, and “CIFS_SERVER\sue” is a member of “CIFS_SERVER\g1” group:

```
cluster1::> vserver cifs users-and-groups local-user show-membership
-vserver vs1
Vserver    User Name                    Membership
---------- ---------------------------- ----------------------
vs1        CIFS_SERVER\Administrator    BUILTIN\Administrators
           CIFS_SERVER\sue              CIFS_SERVER\g1
```

Delete local user accounts

You can delete local user accounts from your storage virtual machine (SVM) if they are no longer needed for local SMB authentication to the CIFS server or for determining access rights to data contained on your SVM.

About this task

Keep the following in mind when deleting local users:

- The file system is not altered.

  Windows Security Descriptors on files and directories that refer to this user are not adjusted.

- All references to local users are removed from the membership and privileges databases.
- Standard, well-known users such as Administrator cannot be deleted.

Steps

1. Determine the name of the local user account that you want to delete: `vserver cifs users-and-groups local-user show -vserver` *`vserver_name`*
2. Delete the local user: `vserver cifs users-and-groups local-user delete -vserver` *`vserver_name`* `-user-name` *`username_name`*
3. Verify that the user account is deleted: `vserver cifs users-and-groups local-user show -vserver` *`vserver_name`*

Example

The following example deletes the local user “CIFS_SERVER\sue” associated with SVM vs1:

```
cluster1::> vserver cifs users-and-groups local-user show -vserver vs1
Vserver  User Name                    Full Name      Description
-------- --------------------------- -------------- -------------
vs1      CIFS_SERVER\Administrator   James Smith    Built-in administrator
account
vs1      CIFS_SERVER\sue             Sue   Jones


cluster1::> vserver cifs users-and-groups local-user delete -vserver vs1
-user-name CIFS_SERVER\sue

cluster1::> vserver cifs users-and-groups local-user show -vserver vs1
Vserver      User Name                    Full Name      Description
-------- --------------------------- -------------- -------------
vs1      CIFS_SERVER\Administrator   James Smith    Built-in administrator
account
```

Manage local groups

Modify local groups

You can modify existing local groups by changing the description for an existing local group or by renaming the group.

If you want to…	Use the command…
Modify the local group description	`vserver cifs users-and-groups local-group modify -vserver` *`vserver_name`* `-group-name` *`group_name`* `-description text` If the description contains a space, then it must be enclosed within double quotation marks.
Rename the local group	`vserver cifs users-and-groups local-group rename -vserver` *`vserver_name`* `-group-name` *`group_name`* `-new-group-name` *`new_group_name`*

Examples

The following example renames the local group “CIFS_SERVER\engineering” to “CIFS_SERVER\engineering_new”:

```
cluster1::> vserver cifs users-and-groups local-group rename -vserver vs1
-group-name CIFS_SERVER\engineering -new-group-name
CIFS_SERVER\engineering_new
```

The following example modifies the description of the local group “CIFS_SERVER\engineering”:

```
cluster1::> vserver cifs users-and-groups local-group modify -vserver vs1
-group-name CIFS_SERVER\engineering -description "New Description"
```

Display information about local groups

You can display a list of all local groups configured on the cluster or on a specified storage virtual machine (SVM). This information can be useful when troubleshooting file-access issues to data contained on the SVM or user-rights (privilege) issues on the SVM.

Step

1. Perform one of the following actions:

If you want information about…	Enter the command…
All local groups on the cluster	`vserver cifs users-and-groups local-group show`
All local groups on the SVM	`vserver cifs users-and-groups local-group show -vserver` *`vserver_name`*

There are other optional parameters that you can choose when you run this command. See the man page for more information.

Example

The following example displays information about all local groups on SVM vs1:

```
cluster1::> vserver cifs users-and-groups local-group show -vserver vs1
Vserver  Group Name                  Description
-------- --------------------------- ----------------------------
vs1      BUILTIN\Administrators      Built-in Administrators group
vs1      BUILTIN\Backup Operators    Backup Operators group
vs1      BUILTIN\Power Users         Restricted administrative privileges
vs1      BUILTIN\Users               All users
vs1      CIFS_SERVER\engineering
vs1      CIFS_SERVER\sales
```

Manage local group membership

You can manage local group membership by adding and removing local or domain users, or adding and removing domain groups. This is useful if you want to control access to data based on access controls placed on the group or if you want users to have privileges associated with that group.

About this task

Guidelines for adding members to a local group:

- You cannot add users to the special *Everyone* group.
- The local group must exist before you can add a user to it.
- The user must exist before you can add the user to a local group.
- You cannot add a local group to another local group.
- To add a domain user or group to a local group, Data ONTAP must be able to resolve the name to a SID.

Guidelines for removing members from a local group:

- You cannot remove members from the special *Everyone* group.
- The group from which you want to remove a member must exist.
- ONTAP must be able to resolve the names of members that you want to remove from the group to a corresponding SID.

Step

1. Add or remove a member in a group.

If you want to…	Then use the command…
Add a member to a group	`vserver cifs users-and-groups local-group add-members -vserver _vserver_name_ -group-name _group_name_ -member-names name[,...]` You can specify a comma-delimited list of local users, domain users, or domain groups to add to the specified local group.
Remove a member from a group	`vserver cifs users-and-groups local-group remove-members -vserver _vserver_name_ -group-name _group_name_ -member-names name[,...]` You can specify a comma-delimited list of local users, domain users, or domain groups to remove from the specified local group.

The following example adds a local user “SMB_SERVER\sue” and a domain group “AD_DOM\dom_eng” to the local group “SMB_SERVER\engineering” on SVM vs1:

```
cluster1::> vserver cifs users-and-groups local-group add-members
-vserver vs1 -group-name SMB_SERVER\engineering -member-names
SMB_SERVER\sue,AD_DOMAIN\dom_eng
```

The following example removes the local users “SMB_SERVER\sue” and “SMB_SERVER\james” from the local group “SMB_SERVER\engineering” on SVM vs1:

```
cluster1::> vserver cifs users-and-groups local-group remove-members
-vserver vs1 -group-name SMB_SERVER\engineering -member-names
SMB_SERVER\sue,SMB_SERVER\james
```

Related information

Displaying information about members of local groups

Display information about members of local groups

You can display a list of all members of local groups configured on the cluster or on a specified storage virtual machine (SVM). This information can be useful when troubleshooting file-access issues or user-rights (privilege) issues.

Step

1. Perform one of the following actions:

If you want to display information about…	Enter the command…
Members of all local groups on the cluster	`vserver cifs users-and-groups local-group show-members`
Members of all local groups on the SVM	`vserver cifs users-and-groups local-group show-members -vserver` *`vserver_name`*

Example

The following example displays information about members of all local groups on SVM vs1:

```
cluster1::> vserver cifs users-and-groups local-group show-members
-vserver vs1
Vserver   Group Name                    Members
--------- ----------------------------- ------------------------
vs1       BUILTIN\Administrators        CIFS_SERVER\Administrator
                                        AD_DOMAIN\Domain Admins
                                        AD_DOMAIN\dom_grp1
          BUILTIN\Users                 AD_DOMAIN\Domain Users
                                        AD_DOMAIN\dom_usr1
          CIFS_SERVER\engineering       CIFS_SERVER\james
```

Delete a local group

You can delete a local group from the storage virtual machine (SVM) if it is no longer needed for determining access rights to data associated with that SVM or if it is no longer needed for assigning SVM user rights (privileges) to group members.

About this task

Keep the following in mind when deleting local groups:

- The file system is not altered.

  Windows Security Descriptors on files and directories that refer to this group are not adjusted.

- If the group does not exist, an error is returned.
- The special *Everyone* group cannot be deleted.
- Built-in groups such as *BUILTIN\Administrators BUILTIN\Users* cannot be deleted.

Steps

1. Determine the name of the local group that you want to delete by displaying the list of local groups on the SVM: `vserver cifs users-and-groups local-group show -vserver vserver_name`
2. Delete the local group: `vserver cifs users-and-groups local-group delete -vserver` *`vserver_name`* `-group-name` *`group_name`*
3. Verify that the group is deleted: `vserver cifs users-and-groups local-user show -vserver` *`vserver_name`*

Example

The following example deletes the local group “CIFS_SERVER\sales” associated with SVM vs1:

```
cluster1::> vserver cifs users-and-groups local-group show -vserver vs1
Vserver   Group Name                    Description
--------- --------------------------- ----------------------------
vs1       BUILTIN\Administrators       Built-in Administrators group
vs1       BUILTIN\Backup Operators     Backup Operators group
vs1       BUILTIN\Power Users          Restricted administrative
privileges
vs1       BUILTIN\Users                All users
vs1       CIFS_SERVER\engineering
vs1       CIFS_SERVER\sales

cluster1::> vserver cifs users-and-groups local-group delete -vserver vs1
-group-name CIFS_SERVER\sales

cluster1::> vserver cifs users-and-groups local-group show -vserver vs1
Vserver   Group Name                    Description
--------- --------------------------- ----------------------------
vs1       BUILTIN\Administrators       Built-in Administrators group
vs1       BUILTIN\Backup Operators     Backup Operators group
vs1       BUILTIN\Power Users          Restricted administrative
privileges
vs1       BUILTIN\Users                All users
vs1       CIFS_SERVER\engineering
```

Update domain user and group names in local databases

You can add domain users and groups to a CIFS server's local groups. These domain objects are registered in local databases on the cluster. If a domain object is renamed, the local databases must be manually updated.

About this task

You must specify the name of the storage virtual machine (SVM) on which you want to update domain names.

Steps

1. Set the privilege level to advanced: `set -privilege advanced`
2. Perform the appropriate action:

If you want to update domain users and groups and…	Use this command…
Display domain users and groups that successfully updated and that failed to update	`vserver cifs users-and-groups update-names -vserver` *`vserver_name`*
Display domain users and groups that successfully updated	`vserver cifs users-and-groups update-names -vserver` *`vserver_name`* `-display -failed-only false`
Display only the domain users and groups that fail to update	`vserver cifs users-and-groups update-names -vserver` *`vserver_name`* `-display -failed-only true`
Suppress all status information about updates	`vserver cifs users-and-groups update-names -vserver` *`vserver_name`* `-suppress -all-output true`

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

Example

The following example updates the names of domain users and groups associated with storage virtual machine (SVM, formerly known as Vserver) vs1. For the last update, there is a dependent chain of names that needs to be updated:

```
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 users-and-groups update-names -vserver vs1

   Vserver:           vs1
   SID:               S-1-5-21-123456789-234565432-987654321-12345
   Domain:            EXAMPLE1
   Out-of-date Name:  dom_user1
   Updated Name:      dom_user2
   Status:            Successfully updated

   Vserver:           vs1
   SID:               S-1-5-21-123456789-234565432-987654322-23456
   Domain:            EXAMPLE2
   Out-of-date Name:  dom_user1
   Updated Name:      dom_user2
   Status:            Successfully updated

   Vserver:           vs1
   SID:               S-1-5-21-123456789-234565432-987654321-123456
   Domain:            EXAMPLE1
   Out-of-date Name:  dom_user3
   Updated Name:      dom_user4
   Status:            Successfully updated; also updated SID "S-1-5-21-
123456789-234565432-987654321-123457"
                      to name "dom_user5"; also updated SID "S-1-5-21-
123456789-234565432-987654321-123458"
                      to name "dom_user6"; also updated SID "S-1-5-21-
123456789-234565432-987654321-123459"
                      to name "dom_user7"; also updated SID "S-1-5-21-
123456789-234565432-987654321-123460"
                      to name "dom_user8"

The command completed successfully. 7 Active Directory objects have been
updated.

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

Manage local privileges

Add privileges to local or domain users or groups

You can manage user rights for local or domain users or groups by adding privileges. The added privileges override the default privileges assigned to any of these objects. This provides enhanced security by allowing you to customize what privileges a user or group has.

Before you begin

The local or domain user or group to which privileges will be added must already exist.

About this task

Adding a privilege to an object overrides the default privileges for that user or group. Adding a privilege does not remove previously added privileges.

You must keep the following in mind when adding privileges to local or domain users or groups:

- You can add one or more privileges.
- When adding privileges to a domain user or group, ONTAP might validate the domain user or group by contacting the domain controller.

  The command might fail if ONTAP is unable to contact the domain controller.

Steps

1. Add one or more privileges to a local or domain user or group: `vserver cifs users-and-groups privilege add-privilege -vserver _vserver_name_ -user-or-group-name name -privileges _privilege_[,...]`
2. Verify that the desired privileges are applied to the object: `vserver cifs users-and-groups privilege show -vserver` *`vserver_name`* `-user-or-group-name` *`name`*

Example

The following example adds the privileges “SeTcbPrivilege” and “SeTakeOwnershipPrivilege” to the user “CIFS_SERVER\sue” on storage virtual machine (SVM, formerly known as Vserver) vs1:

```
cluster1::> vserver cifs users-and-groups privilege add-privilege -vserver
vs1 -user-or-group-name CIFS_SERVER\sue -privileges
SeTcbPrivilege,SeTakeOwnershipPrivilege

cluster1::> vserver cifs users-and-groups privilege show -vserver vs1
Vserver   User or Group Name    Privileges
--------- -------------------- ---------------
vs1       CIFS_SERVER\sue       SeTcbPrivilege
                                SeTakeOwnershipPrivilege
```

Remove privileges from local or domain users or groups

You can manage user rights for local or domain users or groups by removing privileges. This provides enhanced security by allowing you to customize the maximum privileges

that users and groups have.

Before you begin

The local or domain user or group from which privileges will be removed must already exist.

About this task

You must keep the following in mind when removing privileges from local or domain users or groups:

- You can remove one or more privileges.
- When removing privileges from a domain user or group, ONTAP might validate the domain user or group by contacting the domain controller.

  The command might fail if ONTAP is unable to contact the domain controller.

Steps

1. Remove one or more privileges from a local or domain user or group: `vserver cifs users-and-groups privilege remove-privilege -vserver _vserver_name_ -user-or-group-name _name_ -privileges _privilege_[,...]`
2. Verify that the desired privileges have been removed from the object: `vserver cifs users-and-groups privilege show -vserver` *`vserver_name`* `-user-or-group-name` *`name`*

Example

The following example removes the privileges “SeTcbPrivilege” and “SeTakeOwnershipPrivilege” from the user “CIFS_SERVER\sue” on storage virtual machine (SVM, formerly known as Vserver) vs1:

```
cluster1::> vserver cifs users-and-groups privilege show -vserver vs1
Vserver   User or Group Name    Privileges
--------- -------------------- ---------------
vs1       CIFS_SERVER\sue       SeTcbPrivilege
                                SeTakeOwnershipPrivilege

cluster1::> vserver cifs users-and-groups privilege remove-privilege
-vserver vs1 -user-or-group-name CIFS_SERVER\sue -privileges
SeTcbPrivilege,SeTakeOwnershipPrivilege

cluster1::> vserver cifs users-and-groups privilege show -vserver vs1
Vserver   User or Group Name    Privileges
--------- -------------------- -------------------
vs1       CIFS_SERVER\sue       -
```

Reset privileges for local or domain users and groups

You can reset privileges for local or domain users and groups. This can be useful when you have made modifications to privileges for a local or domain user or group and those modifications are no longer wanted or needed.

About this task

Resetting privileges for a local or domain user or group removes any privilege entries for that object.

Steps

1. Reset the privileges on a local or domain user or group: `vserver cifs users-and-groups privilege reset-privilege -vserver vserver_name -user-or-group-name name`
2. Verify that the privileges are reset on the object: `vserver cifs users-and-groups privilege show -vserver vserver_name -user-or-group-name name`

Examples

The following example resets the privileges on the user “CIFS_SERVER\sue” on storage virtual machine (SVM, formerly known as Vserver) vs1. By default, normal users do not have privileges associated with their accounts:

```
cluster1::> vserver cifs users-and-groups privilege show
Vserver   User or Group Name    Privileges
--------- -------------------- ---------------
vs1       CIFS_SERVER\sue      SeTcbPrivilege
                               SeTakeOwnershipPrivilege

cluster1::> vserver cifs users-and-groups privilege reset-privilege
-vserver vs1 -user-or-group-name CIFS_SERVER\sue

cluster1::> vserver cifs users-and-groups privilege show
This table is currently empty.
```

The following example resets the privileges for the group “BUILTIN\Administrators”, effectively removing the privilege entry:

```
cluster1::> vserver cifs users-and-groups privilege show
Vserver   User or Group Name       Privileges
--------- ---------------------- ------------------
vs1       BUILTIN\Administrators   SeRestorePrivilege
                                   SeSecurityPrivilege
                                   SeTakeOwnershipPrivilege

cluster1::> vserver cifs users-and-groups privilege reset-privilege
-vserver vs1 -user-or-group-name BUILTIN\Administrators

cluster1::> vserver cifs users-and-groups privilege show
This table is currently empty.
```

Display information about privilege overrides

You can display information about custom privileges assigned to domain or local user accounts or groups. This information helps you determine whether the desired user rights

are applied.

Step

1. Perform one of the following actions:

If you want to display information about…	Enter this command…
Custom privileges for all domain and local users and groups on the storage virtual machine (SVM)	`vserver cifs users-and-groups privilege show -vserver vserver_name`
Custom privileges for a specific domain or local user and group on the SVM	`vserver cifs users-and-groups privilege show -vserver vserver_name -user-or-group-name name`

   There are other optional parameters that you can choose when you run this command. See the man page for more information.

Example

The following command displays all privileges explicitly associated with local or domain users and groups for SVM vs1:

```
cluster1::> vserver cifs users-and-groups privilege show -vserver vs1
Vserver    User or Group Name     Privileges
---------- ---------------------  ---------------
vs1        BUILTIN\Administrators  SeTakeOwnershipPrivilege
                                   SeRestorePrivilege
vs1        CIFS_SERVER\sue         SeTcbPrivilege
                                   SeTakeOwnershipPrivilege
```

Configure bypass traverse checking

Configure bypass traverse checking overview

Bypass traverse checking is a user right (also known as a *privilege*) that determines whether a user can traverse all the directories in the path to a file even if the user does not have permissions on the traversed directory. You should understand what happens when allowing or disallowing bypass traverse checking, and how to configure bypass traverse checking for users on storage virtual machines (SVMs).

What happens when allowing or disallowing bypass traverse checking

- If allowed, when a user attempts to access a file, ONTAP does not check the traverse permission for the intermediate directories when determining whether to grant or deny access to the file.
- If disallowed, ONTAP checks the traverse (execute) permission for all directories in the path to the file.

  If any of the intermediate directories do not have the “X” (traverse permission), ONTAP denies access to

the file.

Configure bypass traverse checking

You can configure bypass traverse checking by using the ONTAP CLI or by configuring Active Directory group policies with this user right.

The `SeChangeNotifyPrivilege` privilege controls whether users are allowed to bypass traverse checking.

- Adding it to local SMB users or groups on the SVM or to domain users or groups allows bypass traverse checking.
- Removing it from local SMB users or groups on the SVM or from domain users or groups disallows bypass traverse checking.

By default, the following BUILTIN groups on the SVM have the right to bypass traverse checking:

- `BUILTIN\Administrators`
- `BUILTIN\Power Users`
- `BUILTIN\Backup Operators`
- `BUILTIN\Users`
- `Everyone`

If you do not want to allow members of one of these groups to bypass traverse checking, you must remove this privilege from the group.

You must keep the following in mind when configuring bypass traverse checking for local SMB users and groups on the SVM by using the CLI:

- If you want to allow members of a custom local or domain group to bypass traverse checking, you must add the `SeChangeNotifyPrivilege` privilege to that group.
- If you want to allow an individual local or domain user to bypass traverse checking and that user is not a member of a group with that privilege, you can add the `SeChangeNotifyPrivilege` privilege to that user account.
- You can disable bypass traverse checking for local or domain users or groups by removing the `SeChangeNotifyPrivilege` privilege at any time.

> To disable bypass travers checking for specified local or domain users or groups, you must also remove the `SeChangeNotifyPrivilege` privilege from the `Everyone` group.

Related information

Allow users or groups to bypass directory traverse checking

Disallow users or groups from bypassing directory traverse checking

Configure character mapping for SMB file name translation on volumes

Create SMB share access control lists

Secure file access by using Storage-Level Access Guard

List of supported privileges

Add privileges to local or domain users or groups

Allow users or groups to bypass directory traverse checking

If you want a user to be able traverse all the directories in the path to a file even if the user does not have permissions on a traversed directory, you can add the `SeChangeNotifyPrivilege` privilege to local SMB users or groups on storage virtual machines (SVMs). By default, users are able to bypass directory traverse checking.

Before you begin

- A SMB server must be exist on the SVM.
- The local users and groups SMB server option must be enabled.
- The local or domain user or group to which the `SeChangeNotifyPrivilege` privilege will be added must already exist.

About this task

When adding privileges to a domain user or group, ONTAP might validate the domain user or group by contacting the domain controller. The command might fail if ONTAP cannot contact the domain controller.

Steps

1. Enable bypass traverse checking by adding the `SeChangeNotifyPrivilege` privilege to a local or domain user or group: `vserver cifs users-and-groups privilege add-privilege -vserver` *`vserver_name`* `-user-or-group-name` *`name`* `-privileges SeChangeNotifyPrivilege`

   The value for the `-user-or-group-name` parameter is a local user or group, or a domain user or group.

2. Verify that the specified user or group has bypass traverse checking enabled: `vserver cifs users-and-groups privilege show -vserver` *`vserver_name`* `-user-or-group-name` *`name`*

Example

The following command enables users that belong to the “EXAMPLE\eng” group to bypass directory traverse checking by adding the `SeChangeNotifyPrivilege` privilege to the group:

```
cluster1::> vserver cifs users-and-groups privilege add-privilege -vserver
vs1 -user-or-group-name EXAMPLE\eng -privileges SeChangeNotifyPrivilege

cluster1::> vserver cifs users-and-groups privilege show -vserver vs1
Vserver   User or Group Name    Privileges
--------- -------------------- ---------------
vs1       EXAMPLE\eng          SeChangeNotifyPrivilege
```

Related information

Disallowing users or groups from bypassing directory traverse checking

Disallow users or groups from bypassing directory traverse checking

If you do not want a user to traverse all the directories in the path to a file because the user does not have permissions on the traversed directory, you can remove the `SeChangeNotifyPrivilege` privilege from local SMB users or groups on storage virtual machines (SVMs).

Before you begin

The local or domain user or group from which privileges will be removed must already exist.

About this task

When removing privileges from a domain user or group, ONTAP might validate the domain user or group by contacting the domain controller. The command might fail if ONTAP cannot contact the domain controller.

Steps

1. Disallow bypass traverse checking: `vserver cifs users-and-groups privilege remove-privilege -vserver` *`vserver_name`* `-user-or-group-name` *`name`* `-privileges SeChangeNotifyPrivilege`

   The command removes the `SeChangeNotifyPrivilege` privilege from the local or domain user or group that you specify with the value for the `-user-or-group-name` *`name`* parameter.

2. Verify that the specified user or group has bypass traverse checking disabled: `vserver cifs users-and-groups privilege show -vserver` *`vserver_name`* `-user-or-group-name` *`name`*

Example

The following command disallows users that belong to the “EXAMPLE\eng” group from bypassing directory traverse checking:

```
cluster1::> vserver cifs users-and-groups privilege show -vserver vs1
Vserver   User or Group Name    Privileges
--------- -------------------- ----------------------
vs1       EXAMPLE\eng          SeChangeNotifyPrivilege

cluster1::> vserver cifs users-and-groups privilege remove-privilege
-vserver vs1 -user-or-group-name EXAMPLE\eng -privileges
SeChangeNotifyPrivilege

cluster1::> vserver cifs users-and-groups privilege show -vserver vs1
Vserver   User or Group Name    Privileges
--------- -------------------- ----------------------
vs1       EXAMPLE\eng          -
```

Related information

Allowing users or groups to bypass directory traverse checking

Display information about file security and audit policies

Display information about file security and audit policies overview

You can display information about file security on files and directories contained within volumes on storage virtual machines (SVMs). You can display information about audit policies on FlexVol volumes. If configured, you can display information about Storage-Level Access Guard and Dynamic Access Control security settings on FlexVol volumes.

Displaying information about file security

You can display information about file security applied to data contained within volumes and qtrees (for FlexVol volumes) with the following security styles:

- NTFS
- UNIX
- Mixed

Displaying information about audit policies

You can display information about audit policies for auditing access events on FlexVol volumes over the following NAS protocols:

- SMB (all versions)
- NFSv4.x

Displaying information about Storage-Level Access Guard (SLAG) security

Storage-Level Access Guard security can be applied on FlexVol volumes and qtree objects with the following security styles:

- NTFS
- Mixed
- UNIX (if a CIFS server is configured on the SVM that contains the volume)

Displaying information about Dynamic Access Control (DAC) security

Dynamic Access Control security can be applied on an object within a FlexVol volume with the following security styles:

- NTFS
- Mixed (if the object has NTFS effective security)

Related information

Securing file access by using Storage-Level Access Guard

Displaying information about Storage-Level Access Guard

Display information about file security on NTFS security-style volumes

You can display information about file and directory security on NTFS security-style volumes, including what the security style and effective security styles are, what permissions are applied, and information about DOS attributes. 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.

- Because NTFS security-style volumes and qtrees use only NTFS file permissions and Windows users and groups when determining file access rights, UNIX-related output fields contain display-only UNIX file permission information.
- ACL output is displayed for file and folders with NTFS security.
- Because Storage-Level Access Guard security can be configured on the volume root or qtree, output for a volume or qtree path where Storage-Level Access Guard is configured might display both regular file ACLs and Storage-Level Access Guard ACLs.
- The output also displays information about Dynamic Access Control ACEs if Dynamic Access Control is configured for the given file or directory path.

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`

Examples

The following example displays the security information about the path `/vol4` in SVM vs1:

```
cluster::> vserver security file-directory show -vserver vs1 -path /vol4

                                Vserver: vs1
                              File Path: /vol4
                      File Inode Number: 64
                         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
```

The following example displays the security information with expanded masks about the path `/data/engineering` in SVM vs1:

```
cluster::> vserver security file-directory show -vserver vs1 -path -path
/data/engineering -expand-mask true

               Vserver: vs1
             File Path: /data/engineering
      File Inode Number: 5544
         Security Style: ntfs
        Effective Style: ntfs
         DOS Attributes: 10
 DOS Attributes in Text: ----D---
Expanded Dos Attributes: 0x10
     ...0 .... .... .... = Offline
     .... ..0. .... .... = Sparse
     .... .... 0... .... = Normal
     .... .... ..0. .... = Archive
     .... .... ...1 .... = Directory
     .... .... .... .0.. = System
     .... .... .... ..0. = Hidden
     .... .... .... ...0 = Read Only
```

```
           Unix User Id: 0
          Unix Group Id: 0
         Unix Mode Bits: 777
 Unix Mode Bits in Text: rwxrwxrwx
                   ACLs: NTFS Security Descriptor
                         Control:0x8004

                             1... .... .... .... = Self Relative
                             .0.. .... .... .... = RM Control Valid
                             ..0. .... .... .... = SACL Protected
                             ...0 .... .... .... = DACL Protected
                             .... 0... .... .... = SACL Inherited
                             .... .0.. .... .... = DACL Inherited
                             .... ..0. .... .... = SACL Inherit Required
                             .... ...0 .... .... = DACL Inherit Required
                             .... .... ..0. .... = SACL Defaulted
                             .... .... ...0 .... = SACL Present
                             .... .... .... 0... = DACL Defaulted
                             .... .... .... .1.. = DACL Present
                             .... .... .... ..0. = Group Defaulted
                             .... .... .... ...0 = Owner Defaulted

                         Owner:BUILTIN\Administrators
                         Group:BUILTIN\Administrators
                         DACL - ACEs
                           ALLOW-Everyone-0x1f01ff
                             0... .... .... .... .... .... .... .... =
Generic Read
                             .0.. .... .... .... .... .... .... .... =
Generic Write
                             ..0. .... .... .... .... .... .... .... =
Generic Execute
                             ...0 .... .... .... .... .... .... .... =
Generic All
                             .... ...0 .... .... .... .... .... .... =
System Security
                             .... .... ...1 .... .... .... .... .... =
Synchronize
                             .... .... .... 1... .... .... .... .... =
Write Owner
                             .... .... .... .1.. .... .... .... .... =
Write DAC
                             .... .... .... ..1. .... .... .... .... =
Read Control
                             .... .... .... ...1 .... .... .... .... =
Delete
```

```
                              .... .... .... .... .... ...1 .... .... =
Write Attributes
                              .... .... .... .... .... .... 1... .... =
Read Attributes
                              .... .... .... .... .... .... .1.. .... =
Delete Child
                              .... .... .... .... .... .... ..1. .... =
Execute
                              .... .... .... .... .... .... ...1 .... =
Write EA
                              .... .... .... .... .... .... .... 1... =
Read EA
                              .... .... .... .... .... .... .... .1.. =
Append
                              .... .... .... .... .... .... .... ..1. =
Write
                              .... .... .... .... .... .... .... ...1 =
Read

                            ALLOW-Everyone-0x10000000-OI|CI|IO
                              0... .... .... .... .... .... .... .... =
Generic Read
                              .0.. .... .... .... .... .... .... .... =
Generic Write
                              ..0. .... .... .... .... .... .... .... =
Generic Execute
                              ...1 .... .... .... .... .... .... .... =
Generic All
                              .... ...0 .... .... .... .... .... .... =
System Security
                              .... .... ...0 .... .... .... .... .... =
Synchronize
                              .... .... .... 0... .... .... .... .... =
Write Owner
                              .... .... .... .0.. .... .... .... .... =
Write DAC
                              .... .... .... ..0. .... .... .... .... =
Read Control
                              .... .... .... ...0 .... .... .... .... =
Delete
                              .... .... .... .... .... ...0 .... .... =
Write Attributes
                              .... .... .... .... .... .... 0... .... =
Read Attributes
                              .... .... .... .... .... .... .0.. .... =
Delete Child
```

```
                                .... .... .... .... .... .... ..0. .... =
Execute
                                .... .... .... .... .... .... ...0 .... =
Write EA
                                .... .... .... .... .... .... .... 0... =
Read EA
                                .... .... .... .... .... .... .... .0.. =
Append
                                .... .... .... .... .... .... .... ..0. =
Write
                                .... .... .... .... .... .... .... ...0 =
Read
```

The following example displays security information, including Storage-Level Access Guard security information, for the 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
```

Related information

Displaying information about file security on mixed security-style volumes

Displaying information about file security on UNIX security-style volumes

Display information about file security on mixed security-style volumes

You can display information about file and directory security on mixed security-style volumes, including what the security style and effective security styles are, what permissions are applied, and information about UNIX owners and groups. 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.

- Mixed security-style volumes and qtrees can contain some files and folders that use UNIX file permissions, either mode bits or NFSv4 ACLs, and some files and directories that use NTFS file permissions.
- The top level of a mixed security-style volume can have either UNIX or NTFS effective security.
- ACL output is displayed only for file and folders with NTFS or NFSv4 security.

  This field is empty for files and directories using UNIX security that have only mode bit permissions applied (no NFSv4 ACLs).

- The owner and group output fields in the ACL output apply only in the case of NTFS security descriptors.
- Because Storage-Level Access Guard security can be configured on a mixed security-style volume or qtree even if the effective security style of the volume root or qtree is UNIX, output for a volume or qtree path where Storage-Level Access Guard is configured might display both UNIX file permissions and Storage-Level Access Guard ACLs.
- If the path entered in the command is to data with NTFS effective security, the output also displays information about Dynamic Access Control ACEs if Dynamic Access Control is configured for the given file or directory path.

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`

Examples

The following example displays the security information about the path `/projects` in SVM vs1 in expanded-mask form. This mixed security-style path has UNIX effective security.

```
cluster1::> vserver security file-directory show -vserver vs1 -path
/projects -expand-mask true

               Vserver: vs1
             File Path: /projects
      File Inode Number: 78
         Security Style: mixed
        Effective Style: unix
         DOS Attributes: 10
 DOS Attributes in Text: ----D---
Expanded Dos Attributes: 0x10
     ...0 .... .... .... = Offline
     .... ..0. .... .... = Sparse
     .... .... 0... .... = Normal
     .... .... ..0. .... = Archive
     .... .... ...1 .... = Directory
     .... .... .... .0.. = System
     .... .... .... ..0. = Hidden
     .... .... .... ...0 = Read Only
           Unix User Id: 0
          Unix Group Id: 1
         Unix Mode Bits: 700
 Unix Mode Bits in Text: rwx------
                   ACLs: -
```

The following example displays the security information about the path `/data` in SVM vs1. This mixed security-style path has an NTFS effective security.

```
cluster1::> vserver security file-directory show -vserver vs1 -path /data

                                Vserver: vs1
                              File Path: /data
                      File Inode Number: 544
                         Security Style: mixed
                        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
```

The following example displays the security information about the volume at the path `/datavol5` in SVM vs1. The top level of this mixed security-style 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
                           AUDIT-EXAMPLE\market-0x1f01ff-SA
                         DACL (Applies to Directories):
                           ALLOW-BUILTIN\Administrators-0x1f01ff
                           ALLOW-CREATOR OWNER-0x1f01ff
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-EXAMPLE\engineering-0x1f01ff
                           ALLOW-EXAMPLE\market-0x1f01ff
                         SACL (Applies to Files):
                           AUDIT-EXAMPLE\Domain Users-0x120089-FA
                           AUDIT-EXAMPLE\engineering-0x1f01ff-SA
                           AUDIT-EXAMPLE\market-0x1f01ff-SA
                         DACL (Applies to Files):
                           ALLOW-BUILTIN\Administrators-0x1f01ff
                           ALLOW-CREATOR OWNER-0x1f01ff
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-EXAMPLE\engineering-0x1f01ff
                           ALLOW-EXAMPLE\market-0x1f01ff
```

Related information

Displaying information about file security on NTFS security-style volumes

Displaying information about file security on UNIX security-style volumes

Display information about file security on UNIX security-style volumes

You can display information about file and directory security on UNIX security-style volumes, including what the security styles and effective security styles are, what

permissions are applied, and information about UNIX owners and groups. 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 directory security information you want to display. You can display the output in summary form or as a detailed list.

- UNIX security-style volumes and qtrees use only UNIX file permissions, either mode bits or NFSv4 ACLs when determining file access rights.
- ACL output is displayed only for file and folders with NFSv4 security.

  This field is empty for files and directories using UNIX security that have only mode bit permissions applied (no NFSv4 ACLs).

- The owner and group output fields in the ACL output does not apply in the case of NFSv4 security descriptors.

  They are only meaningful for NTFS security descriptors.

- Because Storage-Level Access Guard security is supported on a UNIX volume or qtree if a CIFS server is configured on the SVM, the output might contain information about Storage-Level Access Guard security applied to the volume or qtree specified in the `-path` parameter.

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`

Examples

The following example displays the security information about the path `/home` in SVM vs1:

```
cluster1::> vserver security file-directory show -vserver vs1 -path /home

                                Vserver: vs1
                              File Path: /home
                      File Inode Number: 9590
                         Security Style: unix
                        Effective Style: unix
                         DOS Attributes: 10
                 DOS Attributes in Text: ----D---
               Expanded Dos Attributes: -
                           Unix User Id: 0
                          Unix Group Id: 1
                         Unix Mode Bits: 700
                 Unix Mode Bits in Text: rwx------
                                   ACLs: -
```

The following example displays the security information about the path `/home` in SVM vs1 in expanded-mask form:

```
cluster1::> vserver security file-directory show -vserver vs1 -path /home
-expand-mask true

                                Vserver: vs1
                              File Path: /home
                      File Inode Number: 9590
                         Security Style: unix
                        Effective Style: unix
                         DOS Attributes: 10
                 DOS Attributes in Text: ----D---
               Expanded Dos Attributes: 0x10
                    ...0 .... .... .... = Offline
                    .... ..0. .... .... = Sparse
                    .... .... 0... .... = Normal
                    .... .... ..0. .... = Archive
                    .... .... ...1 .... = Directory
                    .... .... .... .0.. = System
                    .... .... .... ..0. = Hidden
                    .... .... .... ...0 = Read Only
                          Unix User Id: 0
                         Unix Group Id: 1
                        Unix Mode Bits: 700
                 Unix Mode Bits in Text: rwx------
                                   ACLs: -
```

Related information

Displaying information about file security on NTFS security-style volumes

Displaying information about file security on mixed security-style volumes

Display information about NTFS audit policies on FlexVol volumes using the CLI

You can display information about NTFS audit policies on FlexVol volumes, including what the security styles and effective security styles are, what permissions are applied, and information about system access control lists. You can use the results to validate your security configuration or to troubleshoot auditing issues.

About this task

You must provide the name of the storage virtual machine (SVM) and the path to the files or folders whose audit information you want to display. You can display the output in summary form or as a detailed list.

- NTFS security-style volumes and qtrees use only NTFS system access control lists (SACLs) for audit policies.
- Files and folders in a mixed security-style volume with NTFS effective security can have NTFS audit policies applied to them.

  Mixed security-style volumes and qtrees can contain some files and directories that use UNIX file permissions, either mode bits or NFSv4 ACLs, and some files and directories that use NTFS file permissions.

- The top level of a mixed security-style volume can have either UNIX or NTFS effective security and might or might not contain NTFS SACLs.
- Because Storage-Level Access Guard security can be configured on a mixed security-style volume or qtree even if the effective security style of the volume root or qtree is UNIX, the output for a volume or qtree path where Storage-Level Access Guard is configured might display both regular file and folder NFSv4 SACLs and Storage-Level Access Guard NTFS SACLs.
- If the path that is entered in the command is to data with NTFS effective security, the output also displays information about Dynamic Access Control ACEs if Dynamic Access Control is configured for the given file or directory path.
- When displaying security information about files and folders with NTFS effective security, UNIX-related output fields contain display-only UNIX file permission information.

  NTFS security-style files and folders use only NTFS file permissions and Windows users and groups when determining file access rights.

- ACL output is displayed only for files and folders with NTFS or NFSv4 security.

  This field is empty for files and folders using UNIX security that have only mode bit permissions applied (no NFSv4 ACLs).

- The owner and group output fields in the ACL output apply only in the case of NTFS security descriptors.

Step

1. Display file and directory audit policy 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`
As a detailed list	`vserver security file-directory show -vserver vserver_name -path path -expand-mask true`

Examples

The following example displays the audit policy information for the path `/corp` in SVM vs1. The path has NTFS effective security. The NTFS security descriptor contains both a SUCCESS and a SUCCESS/FAIL SACL entry.

```
cluster::> vserver security file-directory show -vserver vs1 -path /corp
                Vserver: vs1
              File Path: /corp
      File Inode Number: 357
         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:0x8014
                         Owner:DOMAIN\Administrator
                         Group:BUILTIN\Administrators
                         SACL - ACEs
                           ALL-DOMAIN\Administrator-0x100081-OI|CI|SA|FA
                           SUCCESSFUL-DOMAIN\user1-0x100116-OI|CI|SA
                         DACL - ACEs
                           ALLOW-BUILTIN\Administrators-0x1f01ff-OI|CI
                           ALLOW-BUILTIN\Users-0x1f01ff-OI|CI
                           ALLOW-CREATOR OWNER-0x1f01ff-OI|CI
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff-OI|CI
```

The following example displays the audit policy information for the path `/datavol1` in SVM vs1. The path contains both regular file and folder SACLs and Storage-Level Access Guard SACLs.

```
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:0xaa14
                         Owner:BUILTIN\Administrators
                         Group:BUILTIN\Administrators
                         SACL - ACEs
                           AUDIT-EXAMPLE\marketing-0xf01ff-OI|CI|FA
                         DACL - ACEs
                           ALLOW-EXAMPLE\Domain Admins-0x1f01ff-OI|CI
                           ALLOW-EXAMPLE\marketing-0x1200a9-OI|CI

                         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
```

Display information about NFSv4 audit policies on FlexVol volumes using the CLI

You can display information about NFSv4 audit policies on FlexVol volumes using the ONTAP CLI, including what the security styles and effective security styles are, what

permissions are applied, and information about system access control lists (SACLs). You can use the results to validate your security configuration or to troubleshoot auditing issues.

About this task

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

- UNIX security-style volumes and qtrees use only NFSv4 SACLs for audit policies.
- Files and directories in a mixed security-style volume that are of UNIX security style can have NFSv4 audit policies applied to them.

  Mixed security-style volumes and qtrees can contain some files and directories that use UNIX file permissions, either mode bits or NFSv4 ACLs, and some files and directories that use NTFS file permissions.

- The top level of a mixed security-style volume can have either UNIX or NTFS effective security and might or might not contain NFSv4 SACLs.
- ACL output is displayed only for file and folders with NTFS or NFSv4 security.

  This field is empty for files and folders using UNIX security that have only mode bit permissions applied (no NFSv4 ACLs).

- The owner and group output fields in the ACL output apply only in the case of NTFS security descriptors.
- Because Storage-Level Access Guard security can be configured on a mixed security-style volume or qtree even if the effective security style of the volume root or qtree is UNIX, output for a volume or qtree path where Storage-Level Access Guard is configured might display both regular NFSv4 file and directory SACLs and Storage-Level Access Guard NTFS SACLs.
- Because Storage-Level Access Guard security is supported on a UNIX volume or qtree if a CIFS server is configured on the SVM, the output might contain information about Storage-Level Access Guard security applied to the volume or qtree specified in the `-path` parameter.

Steps

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`

Examples

The following example displays the security information about the path `/lab` in SVM vs1. This UNIX security-style path has an NFSv4 SACL.

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

                Vserver: vs1
              File Path: /lab
      File Inode Number: 288
         Security Style: unix
        Effective Style: unix
         DOS Attributes: 11
 DOS Attributes in Text: ----D--R
Expanded Dos Attributes: -
           Unix User Id: 0
          Unix Group Id: 0
         Unix Mode Bits: 0
 Unix Mode Bits in Text: ---------
                   ACLs: NFSV4 Security Descriptor
                         Control:0x8014
                         SACL - ACEs
                           SUCCESSFUL-S-1-520-0-0xf01ff-SA
                           FAILED-S-1-520-0-0xf01ff-FA
                         DACL - ACEs
                           ALLOW-S-1-520-1-0xf01ff
```

Ways to display information about file security and audit policies

You can use the wildcard character (*) to display information about file security and audit policies of all files and directories under a given path or a root volume.

The wildcard character (**) can be used as the last subcomponent of a given directory path below which you want to display information of all files and directories. If you want to display information of a particular file or directory named as “**”, then you need to provide the complete path inside double quotes ("``").

Example

The following command with the wildcard character displays the information about all files and directories below the path `/1/` of SVM vs1:

```
cluster::> vserver security file-directory show -vserver vs1 –path /1/*

                    Vserver: vs1
                  File Path: /1/1
            Security Style: mixed
           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:0x8514
                             Owner:BUILTIN\Administrators
                             Group:BUILTIN\Administrators
                             DACL - ACEs
                             ALLOW-Everyone-0x1f01ff-OI|CI (Inherited)
                    Vserver: vs1
                  File Path: /1/1/abc
            Security Style: mixed
           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:0x8404
                             Owner:BUILTIN\Administrators
                             Group:BUILTIN\Administrators
                             DACL - ACEs
                             ALLOW-Everyone-0x1f01ff-OI|CI (Inherited)
```

The following command displays the information of a file named as "*" under the path `/vol1/a` of SVM vs1. The path is enclosed within double quotes (" ").

```
cluster::> vserver security file-directory show -vserver vs1 -path
"/vol1/a/*"

                Vserver: vs1
              File Path: "/vol1/a/*"
          Security Style: mixed
         Effective Style: unix
          DOS Attributes: 10
  DOS Attributes in Text: ----D---
Expanded Dos Attributes: -
            Unix User Id: 1002
           Unix Group Id: 65533
          Unix Mode Bits: 755
  Unix Mode Bits in Text: rwxr-xr-x
                    ACLs: NFSV4 Security Descriptor
                          Control:0x8014
                          SACL - ACEs
                            AUDIT-EVERYONE@-0x1f01bf-FI|DI|SA|FA
                          DACL - ACEs
                            ALLOW-EVERYONE@-0x1f00a9-FI|DI
                            ALLOW-OWNER@-0x1f01ff-FI|DI
                            ALLOW-GROUP@-0x1200a9-IG
```

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

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

You can manage NTFS file security, NTFS audit policies, and Storage-Level Access Guard on storage virtual machines (SVMs) by using the CLI.

You can manage NTFS file security and audit policies from SMB clients or by using the CLI. However, using the CLI to configure file security and audit policies removes the need to use a remote client to manage file security. Using the CLI can significantly reduce the time it takes to apply security on many files and folders using a single command.

You can configure Storage-Level Access Guard, which is another layer of security applied by ONTAP to SVM volumes. Storage-Level Access Guard applies to accesses from all NAS protocols to the storage object to which Storage-Level Access Guard is applied.

Storage-Level Access Guard can be configured and managed only from the ONTAP CLI. You cannot manage Storage-Level Access Guard settings from SMB clients. Moreover, if you view the security settings on a file or directory from an NFS or SMB client, you will not see the Storage-Level Access Guard security. Storage-Level Access Guard security cannot be revoked from a client, even by a system (Windows or UNIX) administrator. Therefore, Storage-Level Access Guard provides an extra layer of security for data access that is independently set and managed by the storage administrator.

Even though only NTFS access permissions are supported for Storage-Level Access Guard, ONTAP can perform security checks for access over NFS to data on volumes where Storage-Level Access Guard is applied if the UNIX user maps to a Windows user on the SVM that owns the volume.

NTFS security-style volumes

All files and folders contained within NTFS security-style volumes and qtrees have NTFS effective security. You can use the `vserver security file-directory` command family to implement the following types of security on NTFS security-style volumes:

- File permissions and audit policies to files and folders contained in the volume
- Storage-Level Access Guard security on volumes

Mixed security-style volumes

Mixed security-style volumes and qtrees can contain some files and folders that have UNIX effective security and use UNIX file permissions, either mode bits or NFSv4.x ACLs and NFSv4.x audit policies, and some files and folders that have NTFS effective security and use NTFS file permissions and audit policies. You can use the `vserver security file-directory` command family to apply the following types of security to mixed security-style data:

- File permissions and audit policies to files and folders with NTFS effective security-style in the mixed volume or qtree
- Storage-Level Access Guard to volumes with either NTFS and UNIX effective security-style

UNIX security-style volumes

UNIX security-style volumes and qtrees contain files and folders that have UNIX effective security (either mode bits or NFSv4.x ACLs). You must keep the following in mind if you want to use the `vserver security file-directory` command family to implement security on UNIX security-style volumes:

- The `vserver security file-directory` command family cannot be used to manage UNIX file security and audit policies on UNIX security-style volumes and qtrees.
- You can use the `vserver security file-directory` command family to configure Storage-Level Access Guard on UNIX security-style volumes, provided the SVM with the target volume contains a CIFS server.

Related information

Display information about file security and audit policies

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

Configure and apply audit policies to NTFS files and folders using the CLI

Secure file access by using Storage-Level Access Guard

Use cases for using the CLI to set file and folder security

Because you can apply and manage file and folder security locally without involvement from a remote client, you can significantly reduce the time it takes to set bulk security on

a large number of files or folders.

You can benefit from using the CLI to set file and folder security in the following use cases:

- Storage of files in large enterprise environments, such as file storage in home directories
- Migration of data
- Change of Windows domain
- Standardization of file security and audit policies across NTFS file systems

Limits when using the CLI to set file and folder security

You need to be aware of certain limits when using the CLI to set file and folder security.

- The `vserver security file-directory` command family does not support setting NFSv4 ACLs.

  You can only apply NTFS security descriptors to NTFS files and folders.

How security descriptors are used to apply file and folder security

Security descriptors contain the access control lists that determine what actions a user can perform on files and folders, and what is audited when a user accesses files and folders.

- **Permissions**

  Permissions are allowed or denied by an object's owner and determine what actions an object (users, groups, or computer objects) can perform on specified files or folders.

- **Security descriptors**

  Security descriptors are data structures that contain security information that define permissions associated with a file or folder.

- **Access control lists (ACLs)**

  Access control lists are the lists contained within a security descriptor that contain information on what actions users, groups, or computer objects can perform on the file or folder to which the security descriptor is applied. The security descriptor can contain the following two types of ACLs:

  - Discretionary access control lists (DACLs)
  - System access control lists (SACLs)

- **Discretionary access control lists (DACLs)**

  DACLs contain the list of SIDS for the users, groups, and computer objects who are allowed or denied access to perform actions on files or folders. DACLs contain zero or more access control entries (ACEs).

- **System access control lists (SACLs)**

  SACLs contain the list of SIDS for the users, groups, and computer objects for which successful or failed auditing events are logged. SACLs contain zero or more access control entries (ACEs).

- **Access Control Entries (ACEs)**

  ACEs are individual entries in either DACLs or SACLs:

  - A DACL access control entry specifies the access rights that are allowed or denied for particular users, groups, or computer objects.
  - A SACL access control entry specifies the success or failure events to log when auditing specified actions performed by particular users, groups, or computer objects.

- **Permission inheritance**

  Permission inheritance describes how permissions defined in security descriptors are propagated to an object from a parent object. Only inheritable permissions are inherited by child objects. When setting permissions on the parent object, you can decide whether folders, sub-folders, and files can inherit them with "Apply to `this-folder`, `sub-folders`, and `files`".

Related information

SMB and NFS auditing and security tracing

Configuring and applying audit policies to NTFS files and folders using the CLI

Guidelines for applying file-directory policies that use local users or groups on the SVM disaster recovery destination

There are certain guidelines that you must keep in mind before applying file-directory policies on the storage virtual machine (SVM) disaster recovery destination in an ID discard configuration if your file-directory policy configuration uses local users or groups in either the security descriptor or the DACL or SACL entries.

You can configure a disaster recovery configuration for an SVM where the source SVM on the source cluster replicates the data and configuration from the source SVM to a destination SVM on a destination cluster.

You can set up one of two types of SVM disaster recovery:

- Identity preserved

  With this configuration, the identity of the SVM and the CIFS server is preserved.

- Identity discarded

  With this configuration, the identity of the SVM and the CIFS server is not preserved. In this scenario, the name of the SVM and the CIFS server on the destination SVM is different from the SVM and the CIFS server name on the source SVM.

Guidelines for identity discarded configurations

In an identity discarded configuration, for an SVM source that contains local user, group, and privilege configurations, the name of the local domain (local CIFS server name) must be changed to match the CIFS server name on the SVM destination. For example, if the source SVM name is "vs1" and CIFS server name is "CIFS1", and the destination SVM name is "vs1_dst" and the CIFS server name is "CIFS1_DST", then the local domain name for a local user named "CIFS1\user1" is automatically changed to "CIFS1_DST\user1" on the destination SVM:

```
cluster1::> vserver cifs users-and-groups local-user show -vserver vs1_dst

Vserver      User Name                Full Name      Description
------------ ------------------------ -------------- -------------
vs1          CIFS1\Administrator                     Built-in
administrator account
vs1          CIFS1\user1              -              -

cluster1dst::> vserver cifs users-and-groups local-user show -vserver
vs1_dst

Vserver      User Name                Full Name      Description
------------ ------------------------ -------------- -------------
vs1_dst      CIFS1_DST\Administrator                 Built-in
administrator account
vs1_dst      CIFS1_DST\user1          -              -
```

Even though local user and group names are automatically changed in the local user and group databases, local users or group names are not automatically changed in file-directory policy configurations (policies configured on the CLI using the `vserver security file-directory` command family).

For example, for “vs1”, if you have configured a DACL entry where the `-account` parameter is set to “CIFS1\user1”, the setting is not automatically changed on the destination SVM to reflect the destination’s CIFS server name.

```
cluster1::> vserver security file-directory ntfs dacl show -vserver vs1

Vserver: vs1
  NTFS Security Descriptor Name: sd1

    Account Name     Access   Access             Apply To
                     Type     Rights
    --------------   -------  -------            -----------
    CIFS1\user1      allow    full-control      this-folder

cluster1::> vserver security file-directory ntfs dacl show -vserver
vs1_dst

Vserver: vs1_dst
  NTFS Security Descriptor Name: sd1

    Account Name     Access   Access             Apply To
                     Type     Rights
    --------------   -------  -------            -----------
    **CIFS1**\user1          allow    full-control      this-folder
```

You must use the `vserver security file-directory modify` commands to manually change the CIFS server name to the destination CIFS server name.

File-directory policy configuration components that contain account parameters

There are three file-directory policy configuration components that can use parameter settings that can contain local users or groups:

- Security descriptor

  You can optionally specify the owner of the security descriptor and the primary group of the owner of the security descriptor. If the security descriptor uses a local user or group for the owner and primary group entries, you must modify the security descriptor to use the destination SVM in the account name. You can use the `vserver security file-directory ntfs modify` command to make any necessary changes to the account names.

- DACL entries

  Each DACL entry must be associated with an account. You must modify any DACLs that use local user or group accounts to use the destination SVM name. Because you cannot modify the account name for existing DACL entries, you must remove any DACL entries with local users or groups from the security descriptors, create new DACL entries with the corrected destination account names, and associate these new DACL entries with the appropriate security descriptors.

- SACL entries

  Each SACL entry must be associated with an account. You must modify any SACLs that use local user or group accounts to use the destination SVM name. Because you cannot modify the account name for existing SACL entries, you must remove any SACL entries with local users or groups from the security descriptors, create new SACL entries with the corrected destination account names, and associate these new SACL entries with the appropriate security descriptors.

You must make any necessary changes to local users or groups used in the file-directory policy configuration before applying the policy; otherwise, the apply job fails.

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

Create an NTFS security descriptor

Creating an NTFS security descriptor (file security policy) is the first step in configuring and applying NTFS access control lists (ACLs) to files and folders residing within storage virtual machines (SVMs). You can associate the security descriptor to the file or folder path in a policy task.

About this task

You can create NTFS security descriptors for files and folders residing within NTFS security-style volumes, or for files and folders residing on mixed security-style volumes.

By default, when a security descriptor is created, four discretionary access control list (DACL) access control entries (ACEs) are added to that security descriptor. The four default ACEs are as follows:

Object	Access type	Access rights	Where to apply the permissions
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

You can customize the security descriptor configuration by using the following optional parameters:

- Owner of the security descriptor
- Primary group of the owner
- Raw control flags

The value for any optional parameter is ignored for Storage-Level Access Guard. See the man pages for more information.

Add NTFS DACL access control entries to the NTFS security descriptor

Adding DACL (discretionary access control list) access control entries (ACEs) to the NTFS security descriptor is the second step in configuring and applying NTFS ACLs to a file or folder. Each entry identifies which object is allowed or denied access, and defines what the object can or cannot do to the files or folders defined in the ACE.

About this task

You can add one or more ACEs to the security descriptor's DACL.

If the security descriptor contains a DACL that has existing ACEs, the command adds the new ACE to the DACL. If the security descriptor does not contain a DACL, the command creates the DACL and adds the new ACE to it.

You can optionally customize DACL entries by specifying what rights you want to allow or deny for the account specified in the `-account` parameter. There are three mutually exclusive methods for specifying rights:

- Rights
- Advanced rights
- Raw rights (advanced-privilege)

If you do not specify rights for the DACL entry, the default is to set the rights to `Full Control`.

You can optionally customize DACL entries by specifying how to apply inheritance.

The value for any optional parameter is ignored for Storage-Level Access Guard. See the man pages for more information.

Steps

1. Add a DACL entry to a security descriptor: `vserver security file-directory ntfs dacl add -vserver vserver_name -ntfs-sd SD_name -access-type {allow|deny} -account name_or_SIDoptional_parameters`

   `vserver security file-directory ntfs dacl add -ntfs-sd sd1 -access-type deny -account domain\joe -rights full-control -apply-to this-folder -vserver vs1`

2. Verify that the DACL entry is correct: `vserver security file-directory ntfs dacl show -vserver vserver_name -ntfs-sd SD_name -access-type {allow|deny} -account name_or_SID`

   `vserver security file-directory ntfs dacl show -vserver vs1 -ntfs-sd sd1 -access-type deny -account domain\joe`

   ```
                    Vserver: vs1
   Security Descriptor Name: sd1
                Allow or Deny: deny
            Account Name or SID: DOMAIN\joe
                  Access Rights: full-control
     Advanced Access Rights: -
                       Apply To: this-folder
                  Access Rights: full-control
   ```

Create security policies

Creating a file security policy for SVMs is the third step in configuring and applying ACLs to a file or folder. A policy acts as a container for various tasks, where each task is a single entry that can be applied to files or folders. You can add tasks to the security policy later.

About this task

The tasks that you add to a security policy contain associations between the NTFS security descriptor and the file or folder paths. Therefore, you should associate the security policy with each SVM (containing NTFS security-style volumes or mixed security-style volumes).

Steps

1. Create a security policy: `vserver security file-directory policy create -vserver vserver_name -policy-name policy_name`

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

2. Verify the security policy: `vserver security file-directory policy show`

```
vserver security file-directory policy show
   Vserver          Policy Name
 ------------     --------------
     vs1              policy1
```

Add a task to the security policy

Creating and adding a policy task to a security policy is the fourth step in configuring and applying ACLs to files or folders in SVMs. When you create the policy task, you associate the task with a security policy. You can add one or more task entries to a security policy.

About this task

The security policy is a container for a task. A task refers to a single operation that can be done by a security policy to files or folders with NTFS or mixed security (or to a volume object if configuring Storage-Level Access Guard).

There are two types of tasks:

- File and directory tasks

  Used to specify tasks that apply security descriptors to specified files and folders. ACLs applied through file and directory tasks can be managed with SMB clients or the ONTAP CLI.

- Storage-Level Access Guard tasks

  Used to specify tasks that apply Storage-Level Access Guard security descriptors to a specified volume. ACLs applied through Storage-Level Access Guard tasks can be managed only through the ONTAP CLI.

A task contains definitions for the security configuration of a file (or folder) or set of files (or folders). Every task in a policy is uniquely identified by the path. There can be only one task per path within a single policy. A policy cannot have duplicate task entries.

Guidelines for adding a task to a policy:

- There can be a maximum of 10,000 tasks entries per policy.
- A policy can contain one or more tasks.

  Even though a policy can contain more than one 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.

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

  It will never give extra access permissions.

When adding tasks to security policies, you must specify the following four required parameters:

- SVM name
- Policy name

- Path
- Security descriptor to associate with the path

You can customize the security descriptor configuration by using the following optional parameters:

- Security type
- Propagation mode
- Index position
- Access control type

The value for any optional parameter is ignored for Storage-Level Access Guard. See the man pages for more information.

Steps

1. Add a task with an associated security descriptor to the security policy: `vserver security file-directory policy task add -vserver vserver_name -policy-name policy_name -path path -ntfs-sd SD_nameoptional_parameters`

   `file-directory` is the default value for the `-access-control` parameter. Specifying the access control type when configuring file and directory access tasks is optional.

   ```
   vserver security file-directory policy task add -vserver vs1 -policy-name
   policy1 -path /home/dir1 -security-type ntfs -ntfs-mode propagate -ntfs-sd sd2
   -index-num 1 -access-control file-directory
   ```

2. Verify the policy task configuration: `vserver security file-directory policy task show -vserver vserver_name -policy-name policy_name -path path`

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

   ```
   Vserver: vs1
   Policy: policy1

   Index    File/Folder    Access           Security   NTFS       NTFS
   Security
            Path           Control          Type       Mode
   Descriptor Name
   -----    --------       -----------      --------   ------
   ----------------
   1        /home/dir1     file-directory   ntfs       propagate  sd2
   ```

Apply security policies

Applying a file security policy to SVMs is the last step in creating and applying NTFS ACLs to files or folders.

About this task

You can apply security settings defined in the security policy to NTFS files and folders residing within FlexVol volumes (NTFS or mixed security style).

> When an audit policy and associated SACLs are applied, any existing DACLs are overwritten. When a security policy and its associated DACLs are applied, any existing DACLs are overwritten. You should review existing security policies before creating and applying new ones.

Step

1. Apply a security policy: `vserver security file-directory apply -vserver vserver_name -policy-name policy_name`

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

   The policy apply job is scheduled and the Job ID is returned.

   ```
   [Job 53322]Job is queued: Fsecurity Apply. Use the "Job show 53322 -id
   53322" command to view the status of the operation
   ```

Monitor the security policy job

When applying the security policy to storage virtual machines (SVMs), you can monitor the progress of the task by monitoring the security policy job. This is helpful if you want to ascertain that the application of the security policy succeeded. This is also helpful if you have a long-running job where you are applying bulk security to a large number of files and folders.

About this task

To display detailed information about a security policy job, you should use the `-instance` parameter.

Step

1. Monitor the security policy job: `vserver security file-directory job show -vserver vserver_name`

   ```
   vserver security file-directory job show -vserver vs1
   ```

   ```
   Job ID Name                 Vserver    Node           State
   ------ -------------------- ---------- -------------- ----------
   53322  Fsecurity Apply      vs1        node1          Success
          Description: File Directory Security Apply Job
   ```

Verify the applied file security

You can verify the file security settings to confirm that the files or folders on the storage virtual machine (SVM) to which you applied the security policy have the desired settings.

About this task

You must supply the name of the SVM that contains the data and the path to the file and folders on which you want to verify security settings. You can use the optional `-expand-mask` parameter to display detailed information about the security settings.

Step

1. Display file and folder security settings: `vserver security file-directory show -vserver vserver_name -path path [-expand-mask true]`

   ```
   vserver security file-directory show -vserver vs1 -path /data/engineering
   -expand-mask true
   ```

```
         Vserver: vs1
              File Path: /data/engineering
      File Inode Number: 5544
         Security Style: ntfs
        Effective Style: ntfs
         DOS Attributes: 10
 DOS Attributes in Text: ----D---
Expanded Dos Attributes: 0x10
     ...0 .... .... .... = Offline
     .... ..0. .... .... = Sparse
     .... .... 0... .... = Normal
     .... .... ..0. .... = Archive
     .... .... ...1 .... = Directory
     .... .... .... .0.. = System
     .... .... .... ..0. = Hidden
     .... .... .... ...0 = Read Only
           Unix User Id: 0
          Unix Group Id: 0
         Unix Mode Bits: 777
 Unix Mode Bits in Text: rwxrwxrwx
                   ACLs: NTFS Security Descriptor
                         Control:0x8004

                             1... .... .... .... = Self Relative
                             .0.. .... .... .... = RM Control Valid
                             ..0. .... .... .... = SACL Protected
                             ...0 .... .... .... = DACL Protected
                             .... 0... .... .... = SACL Inherited
                             .... .0.. .... .... = DACL Inherited
                             .... ..0. .... .... = SACL Inherit Required
                             .... ...0 .... .... = DACL Inherit Required
                             .... .... ..0. .... = SACL Defaulted
                             .... .... ...0 .... = SACL Present
                             .... .... .... 0... = DACL Defaulted
                             .... .... .... .1.. = DACL Present
                             .... .... .... ..0. = Group Defaulted
```

```
                            .... .... .... ...0 = Owner Defaulted

                          Owner:BUILTIN\Administrators
                          Group:BUILTIN\Administrators
                          DACL - ACEs
                            ALLOW-Everyone-0x1f01ff
                              0... .... .... .... .... .... .... .... =
Generic Read
                              .0.. .... .... .... .... .... .... .... =
Generic Write
                              ..0. .... .... .... .... .... .... .... =
Generic Execute
                              ...0 .... .... .... .... .... .... .... =
Generic All
                              .... ...0 .... .... .... .... .... .... =
System Security
                              .... .... ...1 .... .... .... .... .... =
Synchronize
                              .... .... .... 1... .... .... .... .... =
Write Owner
                              .... .... .... .1.. .... .... .... .... =
Write DAC
                              .... .... .... ..1. .... .... .... .... =
Read Control
                              .... .... .... ...1 .... .... .... .... =
Delete
                              .... .... .... .... .... ...1 .... .... =
Write Attributes
                              .... .... .... .... .... .... 1... .... =
Read Attributes
                              .... .... .... .... .... .... .1.. .... =
Delete Child
                              .... .... .... .... .... .... ..1. .... =
Execute
                              .... .... .... .... .... .... ...1 .... =
Write EA
                              .... .... .... .... .... .... .... 1... =
Read EA
                              .... .... .... .... .... .... .... .1.. =
Append
                              .... .... .... .... .... .... .... ..1. =
Write
                              .... .... .... .... .... .... .... ...1 =
Read

                            ALLOW-Everyone-0x10000000-OI|CI|IO
```

```
                           0... .... .... .... .... .... .... .... =
Generic Read
                           .0.. .... .... .... .... .... .... .... =
Generic Write
                           ..0. .... .... .... .... .... .... .... =
Generic Execute
                           ...1 .... .... .... .... .... .... .... =
Generic All
                           .... ...0 .... .... .... .... .... .... =
System Security
                           .... .... ...0 .... .... .... .... .... =
Synchronize
                           .... .... .... 0... .... .... .... .... =
Write Owner
                           .... .... .... .0.. .... .... .... .... =
Write DAC
                           .... .... .... ..0. .... .... .... .... =
Read Control
                           .... .... .... ...0 .... .... .... .... =
Delete
                           .... .... .... .... .... ...0 .... .... =
Write Attributes
                           .... .... .... .... .... .... 0... .... =
Read Attributes
                           .... .... .... .... .... .... .0.. .... =
Delete Child
                           .... .... .... .... .... .... ..0. .... =
Execute
                           .... .... .... .... .... .... ...0 .... =
Write EA
                           .... .... .... .... .... .... .... 0... =
Read EA
                           .... .... .... .... .... .... .... .0.. =
Append
                           .... .... .... .... .... .... .... ..0. =
Write
                           .... .... .... .... .... .... .... ...0 =
Read
```

Configure and apply audit policies to NTFS files and folders using the CLI overview

There are several steps you must perform to apply audit policies to NTFS files and folders when using the ONTAP CLI. First, you create an NTFS security descriptor and add SACLs to the security descriptor. Next you create a security policy and add policy tasks. You then apply the security policy to a storage virtual machine (SVM).

About this task

After applying the security policy, you can monitor the security policy job and then verify the settings for the applied audit policy.

When an audit policy and associated SACLs are applied, any existing DACLs are overwritten. You should review existing security policies before creating and applying new ones.

Related information

Securing file access by using Storage-Level Access Guard

Limits when using the CLI to set file and folder security

How security descriptors are used to apply file and folder security

SMB and NFS auditing and security tracing

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

Create an NTFS security descriptor

Creating an NTFS security descriptor audit policy is the first step in configuring and applying NTFS access control lists (ACLs) to files and folders residing within SVMs. You will associate the security descriptor to the file or folder path in a policy task.

About this task

You can create NTFS security descriptors for files and folders residing within NTFS security-style volumes, or for files and folders residing on mixed security-style volumes.

By default, when a security descriptor is created, four discretionary access control list (DACL) access control entries (ACEs) are added to that security descriptor. The four default ACEs are as follows:

Object	Access type	Access rights	Where to apply the permissions
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

You can customize the security descriptor configuration by using the following optional parameters:

- Owner of the security descriptor
- Primary group of the owner

- Raw control flags

The value for any optional parameter is ignored for Storage-Level Access Guard. See the man pages for more information.

Steps

1. If you want to use the advanced parameters, set the privilege level to advanced: `set -privilege advanced`
2. Create a security descriptor: `vserver security file-directory ntfs create -vserver vserver_name -ntfs-sd SD_nameoptional_parameters`

   `vserver security file-directory ntfs create -ntfs-sd sd1 -vserver vs1 -owner DOMAIN\joe`

3. Verify that the security descriptor configuration is correct: `vserver security file-directory ntfs show -vserver vserver_name -ntfs-sd SD_name`

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

   ```
                                   Vserver: vs1
                         Security Descriptor Name: sd1
                  Owner of the Security Descriptor: DOMAIN\joe
   ```

4. If you are in the advanced privilege level, return to the admin privilege level: `set -privilege admin`

Add NTFS SACL access control entries to the NTFS security descriptor

Adding SACL (system access control list) access control entries (ACEs) to the NTFS security descriptor is the second step in creating NTFS audit policies for files or folders in SVMs. Each entry identifies the user or group that you want to audit. The SACL entry defines whether you want to audit successful or failed access attempts.

About this task

You can add one or more ACEs to the security descriptor's SACL.

If the security descriptor contains a SACL that has existing ACEs, the command adds the new ACE to the SACL. If the security descriptor does not contain a SACL, the command creates the SACL and adds the new ACE to it.

You can configure SACL entries by specifying what rights you want to audit for success or failure events for the account specified in the `-account` parameter. There are three mutually exclusive methods for specifying rights:

- Rights
- Advanced rights
- Raw rights (advanced-privilege)

If you do not specify rights for the SACL entry, the default setting is `Full Control`.

You can optionally customize SACL entries by specifying how to apply inheritance with the `apply to` parameter. If you do not specify this parameter, the default is to apply this SACL entry to this folder, subfolders, and files.

Steps

1. Add a SACL entry to a security descriptor: `vserver security file-directory ntfs sacl add -vserver vserver_name -ntfs-sd SD_name -access-type {failure|success} -account name_or_SIDoptional_parameters`

   ```
   vserver security file-directory ntfs sacl add -ntfs-sd sd1 -access-type
   failure -account domain\joe -rights full-control -apply-to this-folder
   -vserver vs1
   ```

2. Verify that the SACL entry is correct: `vserver security file-directory ntfs sacl show -vserver vserver_name -ntfs-sd SD_name -access-type {failure|success} -account name_or_SID`

   ```
   vserver security file-directory ntfs sacl show -vserver vs1 -ntfs-sd sd1
   -access-type deny -account domain\joe
   ```

   ```
                                Vserver: vs1
                  Security Descriptor Name: sd1
       Access type for Specified Access Rights: failure
                       Account Name or SID: DOMAIN\joe
                             Access Rights: full-control
                    Advanced Access Rights: -
                                  Apply To: this-folder
                             Access Rights: full-control
   ```

Create security policies

Creating an audit policy for storage virtual machines (SVMs) is the third step in configuring and applying ACLs to a file or folder. A policy acts as a container for various tasks, where each task is a single entry that can be applied to files or folders. You can add tasks to the security policy later.

About this task

The tasks that you add to a security policy contain associations between the NTFS security descriptor and the file or folder paths. Therefore, you should associate the security policy with each storage virtual machine (SVM) (containing NTFS security-style volumes or mixed security-style volumes).

Steps

1. Create a security policy: `vserver security file-directory policy create -vserver vserver_name -policy-name policy_name`

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

```
vs1
```

2. Verify the security policy: `vserver security file-directory policy show`

```
vserver security file-directory policy show
   Vserver          Policy Name
 ------------     --------------
     vs1              policy1
```

Add a task to the security policy

Creating and adding a policy task to a security policy is the fourth step in configuring and applying ACLs to files or folders in SVMs. When you create the policy task, you associate the task with a security policy. You can add one or more task entries to a security policy.

About this task

The security policy is a container for a task. A task refers to a single operation that can be done by a security policy to files or folders with NTFS or mixed security (or to a volume object if configuring Storage-Level Access Guard).

There are two types of tasks:

- File and directory tasks

  Used to specify tasks that apply security descriptors to specified files and folders. ACLs applied through file and directory tasks can be managed with SMB clients or the ONTAP CLI.

- Storage-Level Access Guard tasks

  Used to specify tasks that apply Storage-Level Access Guard security descriptors to a specified volume. ACLs applied through Storage-Level Access Guard tasks can be managed only through the ONTAP CLI.

A task contains definitions for the security configuration of a file (or folder) or set of files (or folders). Every task in a policy is uniquely identified by the path. There can be only one task per path within a single policy. A policy cannot have duplicate task entries.

Guidelines for adding a task to a policy:

- There can be a maximum of 10,000 tasks entries per policy.
- A policy can contain one or more tasks.

  Even though a policy can contain more than one 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.

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

  It will never give extra access permissions.

You can customize the security descriptor configuration by using the following optional parameters:

- Security type
- Propagation mode
- Index position
- Access control type

The value for any optional parameter is ignored for Storage-Level Access Guard. See the man pages for more information.

Steps

1. Add a task with an associated security descriptor to the security policy: `vserver security file-directory policy task add -vserver vserver_name -policy-name policy_name -path path -ntfs-sd SD_nameoptional_parameters`

   `file-directory` is the default value for the `-access-control` parameter. Specifying the access control type when configuring file and directory access tasks is optional.

   ```
   vserver security file-directory policy task add -vserver vs1 -policy-name
   policy1 -path /home/dir1 -security-type ntfs -ntfs-mode propagate -ntfs-sd sd2
   -index-num 1 -access-control file-directory
   ```

2. Verify the policy task configuration: `vserver security file-directory policy task show -vserver vserver_name -policy-name policy_name -path path`

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

   ```
   Vserver: vs1
   Policy: policy1

   Index    File/Folder    Access           Security   NTFS       NTFS
   Security
            Path           Control          Type       Mode
   Descriptor Name
   -----    --------       -----------      --------   ------
   ----------------
   1        /home/dir1     file-directory   ntfs       propagate  sd2
   ```

Apply security policies

Applying an audit policy to SVMs is the last step in creating and applying NTFS ACLs to files or folders.

About this task

You can apply security settings defined in the security policy to NTFS files and folders residing within FlexVol volumes (NTFS or mixed security style).

When an audit policy and associated SACLs are applied, any existing DACLs are overwritten. When a security policy and its associated DACLs are applied, any existing DACLs are overwritten. You should review existing security policies before creating and applying new ones.

Step

1. Apply a security policy: `vserver security file-directory apply -vserver vserver_name -policy-name policy_name`

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

   The policy apply job is scheduled and the Job ID is returned.

   ```
   [Job 53322]Job is queued: Fsecurity Apply. Use the "Job show 53322 -id
   53322" command to view the status of the operation
   ```

Monitor the security policy job

When applying the security policy to storage virtual machines (SVMs), you can monitor the progress of the task by monitoring the security policy job. This is helpful if you want to ascertain that the application of the security policy succeeded. This is also helpful if you have a long-running job where you are applying bulk security to a large number of files and folders.

About this task

To display detailed information about a security policy job, you should use the `-instance` parameter.

Step

1. Monitor the security policy job: `vserver security file-directory job show -vserver vserver_name`

   ```
   vserver security file-directory job show -vserver vs1
   ```

   ```
   Job ID Name                 Vserver    Node           State
   ------ -------------------- ---------- -------------- ----------
   53322  Fsecurity Apply      vs1        node1          Success
          Description: File Directory Security Apply Job
   ```

Verify the applied audit policy

You can verify the audit policy to confirm that the files or folders on the storage virtual machine (SVM) to which you applied the security policy have the desired audit security settings.

About this task

You use the `vserver security file-directory show` command to display audit policy information. You

must supply the name of the SVM that contains the data and the path to the data whose file or folder audit policy information you want to display.

Step

1. Display audit policy settings: `vserver security file-directory show -vserver vserver_name -path path`

Example

The following command displays the audit policy information applied to the path “/corp” in SVM vs1. The path has both a SUCCESS and a SUCCESS/FAIL SACL entry applied to it:

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

                Vserver: vs1
              File Path: /corp
      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:0x8014
                         Owner:DOMAIN\Administrator
                         Group:BUILTIN\Administrators
                         SACL - ACEs
                           ALL-DOMAIN\Administrator-0x100081-OI|CI|SA|FA
                           SUCCESSFUL-DOMAIN\user1-0x100116-OI|CI|SA
                         DACL - ACEs
                           ALLOW-BUILTIN\Administrators-0x1f01ff-OI|CI
                           ALLOW-BUILTIN\Users-0x1f01ff-OI|CI
                           ALLOW-CREATOR OWNER-0x1f01ff-OI|CI
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff-OI|CI
```

Considerations when managing security policy jobs

If a security policy job exists, under certain circumstances, you cannot modify that security policy or the tasks assigned to that policy. You should understand under what conditions you can or cannot modify security policies so that any attempts that you make to modify the policy are successful. Modifications to the policy include adding, removing, or modifying tasks assigned to the policy and deleting or modifying the policy.

You cannot modify a security policy or a task assigned to that policy if a job exists for that policy and that job is in the following states:

- The job is running or in progress.
- The job is paused.
- The job is resumed and is in the running state.
- If the job is waiting to failover to another node.

Under the following circumstances, if a job exists for a security policy, you can successfully modify that security policy or a task assigned to that policy:

- The policy job is stopped.
- The policy job has successfully finished.

Commands for managing NTFS security descriptors

There are specific ONTAP commands for managing security descriptors. You can create, modify, delete, and display information about security descriptors.

If you want to…	Use this command…
Create NTFS security descriptors	`vserver security file-directory ntfs create`
Modify existing NTFS security descriptors	`vserver security file-directory ntfs modify`
Display information about existing NTFS security descriptors	`vserver security file-directory ntfs show`
Delete NTFS security descriptors	`vserver security file-directory ntfs delete`

See the man pages for the `vserver security file-directory ntfs` commands for more information.

Commands for managing NTFS DACL access control entries

There are specific ONTAP commands for managing DACL access control entries (ACEs). You can add ACEs to NTFS DACLs at any time. You can also manage existing NTFS DACLs by modifying, deleting, and displaying information about ACEs in DACLs.

If you want to…	Use this command…
Create ACEs and add them to NTFS DACLs	`vserver security file-directory ntfs dacl add`
Modify existing ACEs in NTFS DACLs	`vserver security file-directory ntfs dacl modify`

If you want to…	Use this command…
Display information about existing ACEs in NTFS DACLs	`vserver security file-directory ntfs dacl show`
Remove existing ACEs from NTFS DACLs	`vserver security file-directory ntfs dacl remove`

See the man pages for the `vserver security file-directory ntfs dacl` commands for more information.

Commands for managing NTFS SACL access control entries

There are specific ONTAP commands for managing SACL access control entries (ACEs). You can add ACEs to NTFS SACLs at any time. You can also manage existing NTFS SACLs by modifying, deleting, and displaying information about ACEs in SACLs.

If you want to…	Use this command…
Create ACEs and add them to NTFS SACLs	`vserver security file-directory ntfs sacl add`
Modify existing ACEs in NTFS SACLs	`vserver security file-directory ntfs sacl modify`
Display information about existing ACEs in NTFS SACLs	`vserver security file-directory ntfs sacl show`
Remove existing ACEs from NTFS SACLs	`vserver security file-directory ntfs sacl remove`

See the man pages for the `vserver security file-directory ntfs sacl` commands for more information.

Commands for managing security policies

There are specific ONTAP commands for managing security policies. You can display information about policies and you can delete policies. You cannot modify a security policy.

If you want to…	Use this command…
Create security policies	`vserver security file-directory policy create`

If you want to…	Use this command…
Display information about security policies	`vserver security file-directory policy show`
Delete security policies	`vserver security file-directory policy delete`

See the man pages for the `vserver security file-directory policy` commands for more information.

Commands for managing security policy tasks

There are ONTAP commands for adding, modifying, removing, and displaying information about security policy tasks.

If you want to…	Use this command…
Add security policy tasks	`vserver security file-directory policy task add`
Modify security policy tasks	`vserver security file-directory policy task modify`
Display information about security policy tasks	`vserver security file-directory policy task show`
Remove security policy tasks	`vserver security file-directory policy task remove`

See the man pages for the `vserver security file-directory policy task` commands for more information.

Commands for managing security policy jobs

There are ONTAP commands for pausing, resuming, stopping, and displaying information about security policy jobs.

If you want to…	Use this command…
Pause security policy jobs	`vserver security file-directory job pause -vserver vserver_name -id integer`
Resume security policy jobs	`vserver security file-directory job resume -vserver vserver_name -id integer`

If you want to…	Use this command…
Display information about security policy jobs	`vserver security file-directory job show -vserver vserver_name` You can determine the job ID of a job using this command.
Stop security policy jobs	`vserver security file-directory job stop -vserver vserver_name -id integer`

See the man pages for the `vserver security file-directory job` commands for more information.

Configure the metadata cache for SMB shares

How SMB metadata caching works

Metadata caching enables file attribute caching on SMB 1.0 clients to provide faster access to file and folder attributes. You can enable or disable attribute caching on a per-share basis. You can also configure the time-to-live for cached entries if metadata caching is enabled. Configuring metadata caching is not necessary if clients are connecting to shares over SMB 2.x or SMB 3.0.

When enabled, the SMB metadata cache stores path and file attribute data for a limited amount of time. This can improve SMB performance for SMB 1.0 clients with common workloads.

For certain tasks, SMB creates a significant amount of traffic that can include multiple identical queries for path and file metadata. You can reduce the number of redundant queries and improve performance for SMB 1.0 clients by using SMB metadata caching to fetch information from the cache instead.

While unlikely, it is possible that the metadata cache might serve stale information to SMB 1.0 clients. If your environment cannot afford this risk, you should not enable this feature.

Enable the SMB metadata cache

You can improve SMB performance for SMB 1.0 clients by enabling the SMB metadata cache. By default, SMB metadata caching is disabled.

Step

1. Perform the desired action:

If you want to…	Enter the command…
Enable SMB metadata caching when you create a share	`vserver cifs share create -vserver` *`vserver_name`* `-share-name` *`share_name`* `-path` *`path`* `-share-properties attributecache`

If you want to…	Enter the command…
Enable SMB metadata caching on an existing share	`vserver cifs share properties add -vserver vserver_name -share-name share_name -share-properties attributecache`

Related information

Configuring the lifetime of SMB metadata cache entries

Adding or removing share properties on an existing SMB share

Configure the lifetime of SMB metadata cache entries

You can configure the lifetime of SMB metadata cache entries to optimize the SMB metadata cache performance in your environment. The default is 10 seconds.

Before you begin

You must have enabled the SMB metadata cache feature. If SMB metadata caching is not enabled, the SMB cache TTL setting is not used.

Step

1. Perform the desired action:

If you want to configure the lifetime of SMB metadata cache entries when you…	Enter the command…
Create a share	`vserver cifs share -create -vserver vserver_name -share-name share_name -path path -attribute-cache-ttl [integerh][integerm][integers]`
Modify an existing share	`vserver cifs share -modify -vserver vserver_name -share-name share_name -attribute-cache-ttl [integerh][integerm][integers]`

You can specify additional share configuration options and properties when you create or modify shares. See the man pages for more information.

Manage file locks

About file locking between protocols

File locking is a method used by client applications to prevent a user from accessing a file previously opened by another user. How ONTAP locks files depends on the protocol of the client.

If the client is an NFS client, locks are advisory; if the client is an SMB client, locks are mandatory.

Because of differences between the NFS and SMB file locks, an NFS client might fail to access a file previously opened by an SMB application.

The following occurs when an NFS client attempts to access a file locked by an SMB application:

- In mixed or NTFS volumes, file manipulation operations such as `rm`, `rmdir`, and `mv` can cause the NFS application to fail.
- NFS read and write operations are denied by SMB deny-read and deny-write open modes, respectively.
- NFS write operations fail when the written range of the file is locked with an exclusive SMB bytelock.

In UNIX security-style volumes, NFS unlink and rename operations ignore SMB lock state and allow access to the file. All other NFS operations on UNIX security-style volumes honor SMB lock state.

How ONTAP treats read-only bits

The read-only bit is set on a file-by-file basis to reflect whether a file is writable (disabled) or read-only (enabled).

SMB clients that use Windows can set a per-file read-only bit. NFS clients do not set a per-file read-only bit because NFS clients do not have any protocol operations that use a per-file read-only bit.

ONTAP can set a read-only bit on a file when an SMB client that uses Windows creates that file. ONTAP can also set a read-only bit when a file is shared between NFS clients and SMB clients. Some software, when used by NFS clients and SMB clients, requires the read-only bit to be enabled.

For ONTAP to keep the appropriate read and write permissions on a file shared between NFS clients and SMB clients, it treats the read-only bit according to the following rules:

- NFS treats any file with the read-only bit enabled as if it has no write permission bits enabled.
- If an NFS client disables all write permission bits and at least one of those bits had previously been enabled, ONTAP enables the read-only bit for that file.
- If an NFS client enables any write permission bit, ONTAP disables the read-only bit for that file.
- If the read-only bit for a file is enabled and an NFS client attempts to discover permissions for the file, the permission bits for the file are not sent to the NFS client; instead, ONTAP sends the permission bits to the NFS client with the write permission bits masked.
- If the read-only bit for a file is enabled and an SMB client disables the read-only bit, ONTAP enables the owner's write permission bit for the file.
- Files with the read-only bit enabled are writable only by root.

Changes to file permissions take effect immediately on SMB clients, but might not take effect immediately on NFS clients if the NFS client enables attribute caching.

How ONTAP differs from Windows on handling locks on share path components

Unlike Windows, ONTAP does not lock each component of the path to an open file while the file is open. This behavior also affects SMB share paths.

Because ONTAP does not lock each component of the path, it is possible to rename a path component above

the open file or share, which can cause problems for certain applications, or can cause the share path in the SMB configuration to be invalid. This can cause the share to be inaccessible.

To avoid issues caused by renaming path components, you can apply security settings that prevent users or applications from renaming critical directories.

Display information about locks

You can display information about the current file locks, including what types of locks are held and what the lock state is, details about byte-range locks, sharelock modes, delegation locks, and opportunistic locks, and whether locks are opened with durable or persistent handles.

About this task

The client IP address cannot be displayed for locks established through NFSv4 or NFSv4.1.

By default, the command displays information about all locks. You can use command parameters to display information about locks for a specific storage virtual machine (SVM) or to filter the command's output by other criteria.

The `vserver locks show` command displays information about four types of locks:

- Byte-range locks, which lock only a portion of a file.
- Share locks, which lock open files.
- Opportunistic locks, which control client-side caching over SMB.
- Delegations, which control client-side caching over NFSv4.x.

By specifying optional parameters, you can determine important information about each lock type. See the man page for the command for more information.

Step

1. Display information about locks by using the `vserver locks show` command.

Examples

The following example displays summary information for an NFSv4 lock on a file with the path `/vol1/file1`. The sharelock access mode is write-deny_none, and the lock was granted with write delegation:

```
cluster1::> vserver locks show

Vserver: vs0
Volume  Object Path               LIF         Protocol  Lock Type   Client
------- ------------------------ ----------- --------- -----------
-------
vol1    /vol1/file1              lif1        nfsv4     share-level -
               Sharelock Mode: write-deny_none
                                                        delegation  -
               Delegation Type: write
```

The following example displays detailed oplock and sharelock information about the SMB lock on a file with the path `/data2/data2_2/intro.pptx`. A durable handle is granted on the file with a share lock access mode of write-deny_none to a client with an IP address of 10.3.1.3. A lease oplock is granted with a batch oplock level:

```
cluster1::> vserver locks show -instance -path /data2/data2_2/intro.pptx

                    Vserver: vs1
                     Volume: data2_2
          Logical Interface: lif2
                Object Path: /data2/data2_2/intro.pptx
                  Lock UUID: 553cf484-7030-4998-88d3-1125adbba0b7
              Lock Protocol: cifs
                  Lock Type: share-level
   Node Holding Lock State: node3
                 Lock State: granted
  Bytelock Starting Offset: -
    Number of Bytes Locked: -
     Bytelock is Mandatory: -
     Bytelock is Exclusive: -
     Bytelock is Superlock: -
          Bytelock is Soft: -
              Oplock Level: -
   Shared Lock Access Mode: write-deny_none
       Shared Lock is Soft: false
           Delegation Type: -
            Client Address: 10.3.1.3
             SMB Open Type: durable
         SMB Connect State: connected
SMB Expiration Time (Secs): -
         SMB Open Group ID:
78a90c59d45ae211998100059a3c7a00a007f70da0f8ffffcd445b0300000000

                    Vserver: vs1
                     Volume: data2_2
          Logical Interface: lif2
                Object Path: /data2/data2_2/test.pptx
                  Lock UUID: 302fd7b1-f7bf-47ae-9981-f0dcb6a224f9
              Lock Protocol: cifs
                  Lock Type: op-lock
   Node Holding Lock State: node3
                 Lock State: granted
  Bytelock Starting Offset: -
    Number of Bytes Locked: -
     Bytelock is Mandatory: -
     Bytelock is Exclusive: -
```

```
     Bytelock is Superlock: -
          Bytelock is Soft: -
              Oplock Level: batch
   Shared Lock Access Mode: -
       Shared Lock is Soft: -
           Delegation Type: -
            Client Address: 10.3.1.3
             SMB Open Type: -
         SMB Connect State: connected
SMB Expiration Time (Secs): -
         SMB Open Group ID:
78a90c59d45ae211998100059a3c7a00a007f70da0f8ffffcd445b0300000000
```

Break locks

When file locks are preventing client access to files, you can display information about currently held locks, and then break specific locks. Examples of scenarios in which you might need to break locks include debugging applications.

About this task

The `vserver locks break` command is available only at the advanced privilege level and higher. The man page for the command contains detailed information.

Steps

1. To find the information you need to break a lock, use the `vserver locks show` command.

   The man page for the command contains detailed information.

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

If you want to break a lock by specifying…	Enter the command…
The SVM name, volume name, LIF name, and file path	`vserver locks break -vserver vserver_name -volume volume_name -path path -lif lif`
The lock ID	`vserver locks break -lockid UUID`

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

Monitor SMB activity

Display SMB session information

You can display information about established SMB sessions, including the SMB

connection and session ID and the IP address of the workstation using the session. You can display information about the session's SMB protocol version and continuously available protection level, which helps you identify whether the session supports nondisruptive operations.

About this task

You can display information for all of the sessions on your SVM in summary form. However, in many cases, the amount of output that is returned is large. You can customize what information is displayed in the output by specifying optional parameters:

- You can use the optional `-fields` parameter to display output about the fields you choose.

  You can enter `-fields ?` to determine what fields you can use.

- You can use the `-instance` parameter to display detailed information about established SMB sessions.
- You can use the `-fields` parameter or the `-instance` parameter either alone or in combination with other optional parameters.

Step

1. Perform one of the following actions:

If you want to display SMB session information…	Enter the following command…
For all sessions on the SVM in summary form	`vserver cifs session show -vserver vserver_name`
On a specified connection ID	`vserver cifs session show -vserver vserver_name -connection-id integer`
From a specified workstation IP address	`vserver cifs session show -vserver vserver_name -address workstation_IP_address`
On a specified LIF IP address	`vserver cifs session show -vserver vserver_name -lif-address LIF_IP_address`
On a specified node	`vserver cifs session show -vserver vserver_name -node {node_name|local}`
From a specified Windows user	`vserver cifs session show -vserver vserver_name -windows-user domain_name\\user_name`
With a specified authentication mechanism	`vserver cifs session show -vserver vserver_name -auth-mechanism {NTLMv1|NTLMv2|Kerberos|Anonymous}`

If you want to display SMB session information…	Enter the following command…
With a specified protocol version	`vserver cifs session show -vserver vserver_name -protocol-version {SMB1|SMB2|SMB2_1|SMB3|SMB3_1}` Continuously available protection and SMB Multichannel are available only on SMB 3.0 and later sessions. To view their status on all qualifying sessions, you should specify this parameter with the value set to `SMB3` or later.
With a specified level of continuously available protection	`vserver cifs session show -vserver vserver_name -continuously-available {No|Yes|Partial}` If the continuously available status is `Partial`, this means that the session contains at least one open continuously available file, but the session has some files that are not open with continuously available protection. You can use the `vserver cifs sessions file show` command to determine which files on the established session are not open with continuously available protection.
With a specified SMB signing session status	`vserver cifs session show -vserver vserver_name -is-session-signed {true|false}`

Examples

The following command displays session information for the sessions on SVM vs1 established from a workstation with IP address 10.1.1.1:

```
cluster1::> vserver cifs session show -address 10.1.1.1
Node:    node1
Vserver: vs1
Connection Session                                    Open         Idle
ID         ID      Workstation      Windows User    Files        Time
---------- ------- ---------------- ------------- ------- ------------
3151272279,
3151272280,
3151272281  1       10.1.1.1         DOMAIN\joe          2          23s
```

The following command displays detailed session information for sessions with continuously available protection on SVM vs1. The connection was made by using the domain account.

```
cluster1::> vserver cifs session show -instance -continuously-available
Yes

                      Node: node1
                   Vserver: vs1
                Session ID: 1
              Connection ID: 3151274158
Incoming Data LIF IP Address: 10.2.1.1
      Workstation IP address: 10.1.1.2
    Authentication Mechanism: Kerberos
                Windows User: DOMAIN\SERVER1$
                   UNIX User: pcuser
                 Open Shares: 1
                  Open Files: 1
                  Open Other: 0
             Connected Time: 10m 43s
                   Idle Time: 1m 19s
           Protocol Version: SMB3
      Continuously Available: Yes
          Is Session Signed: false
       User Authenticated as: domain-user
                NetBIOS Name: -
       SMB Encryption Status: Unencrypted
```

The following command displays session information on a session using SMB 3.0 and SMB Multichannel on SVM vs1. In the example, the user connected to this share from an SMB 3.0 capable client by using the LIF IP address; therefore, the authentication mechanism defaulted to NTLMv2. The connection must be made by using Kerberos authentication to connect with continuously available protection.

```
cluster1::> vserver cifs session show -instance -protocol-version SMB3

                        Node: node1
                     Vserver: vs1
                  Session ID: 1
              **Connection IDs: 3151272607,31512726078,3151272609
            Connection Count: 3**
Incoming Data LIF IP Address: 10.2.1.2
      Workstation IP address: 10.1.1.3
    Authentication Mechanism: NTLMv2
                Windows User: DOMAIN\administrator
                   UNIX User: pcuser
                 Open Shares: 1
                  Open Files: 0
                  Open Other: 0
              Connected Time: 6m 22s
                   Idle Time: 5m 42s
           Protocol Version: SMB3
      Continuously Available: No
           Is Session Signed: false
       User Authenticated as: domain-user
                NetBIOS Name: -
       SMB Encryption Status: Unencrypted
```

Related information

Displaying information about open SMB files

Display information about open SMB files

You can display information about open SMB files, including the SMB connection and session ID, the hosting volume, the share name, and the share path. You can display information about a file's continuously available protection level, which is helpful in determining whether an open file is in a state that supports nondisruptive operations.

About this task

You can display information about open files on an established SMB session. The displayed information is useful when you need to determine SMB session information for particular files within an SMB session.

For example, if you have an SMB session where some of the open files are open with continuously available protection and some are not open with continuously available protection (the value for the `-continuously-available` field in `vserver cifs session show` command output is `Partial`), you can determine which files are not continuously available by using this command.

You can display information for all open files on established SMB sessions on storage virtual machines (SVMs) in summary form by using the `vserver cifs session file show` command without any optional parameters.

However, in many cases, the amount of output returned is large. You can customize what information is displayed in the output by specifying optional parameters. This can be helpful when you want to view information for only a small subset of open files.

- You can use the optional `-fields` parameter to display output on the fields you choose.

  You can use this parameter either alone or in combination with other optional parameters.

- You can use the `-instance` parameter to display detailed information about open SMB files.

  You can use this parameter either alone or in combination with other optional parameters.

Step

1. Perform one of the following actions:

If you want to display open SMB files…	Enter the following command…
On the SVM in summary form	`vserver cifs session file show -vserver vserver_name`
On a specified node	`vserver cifs session file show -vserver vserver_name -node {node_name|local}`
On a specified file ID	`vserver cifs session file show -vserver vserver_name -file-id integer`
On a specified SMB connection ID	`vserver cifs session file show -vserver vserver_name -connection-id integer`
On a specified SMB session ID	`vserver cifs session file show -vserver vserver_name -session-id integer`
On the specified hosting aggregate	`vserver cifs session file show -vserver vserver_name -hosting -aggregate aggregate_name`
On the specified volume	`vserver cifs session file show -vserver vserver_name -hosting-volume volume_name`
On the specified SMB share	`vserver cifs session file show -vserver vserver_name -share share_name`

If you want to display open SMB files…	Enter the following command…
On the specified SMB path	`vserver cifs session file show -vserver vserver_name -path path`
With the specified level of continuously available protection	`vserver cifs session file show -vserver vserver_name -continuously -available {No|Yes}` If the continuously available status is `No`, this means that these open files are not capable of nondisruptively recovering from takeover and giveback. They also cannot recover from general aggregate relocation between partners in a high-availability relationship.
With the specified reconnected state	`vserver cifs session file show -vserver vserver_name -reconnected {No|Yes}` If the reconnected state is `No`, the open file is not reconnected after a disconnection event. This can mean that the file was never disconnected, or that the file was disconnected and is not successfully reconnected. If the reconnected state is `Yes`, this means that the open file is successfully reconnected after a disconnection event.

There are additional optional parameters that you can use to refine the output results. See the man page for more information.

Examples

The following example displays information about open files on SVM vs1:

```
cluster1::> vserver cifs session file show -vserver vs1
Node:       node1
Vserver:    vs1
Connection: 3151274158
Session:    1
File    File      Open Hosting              Continuously
ID      Type      Mode Volume    Share      Available
------- --------- ---- --------- ---------- ------------
41      Regular   r    data      data       Yes
Path: \mytest.rtf
```

The following example displays detailed information about open SMB files with file ID 82 on SVM vs1:

```
cluster1::> vserver cifs session file show -vserver vs1 -file-id 82
-instance

                 Node: node1
              Vserver: vs1
              File ID: 82
        Connection ID: 104617
           Session ID: 1
            File Type: Regular
            Open Mode: rw
Aggregate Hosting File: aggr1
   Volume Hosting File: data1
           CIFS Share: data1
  Path from CIFS Share: windows\win8\test\test.txt
           Share Mode: rw
          Range Locks: 1
Continuously Available: Yes
           Reconnected: No
```

Related information

Displaying SMB session information

Determine which statistics objects and counters are available

Before you can obtain information about CIFS, SMB, auditing, and BranchCache hash statistics and monitor performance, you must know which objects and counters are available from which you can obtain data.

Steps

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

If you want to determine…	Enter…
Which objects are available	`statistics catalog object show`
Specific objects that are available	`statistics catalog object show object object_name`
Which counters are available	`statistics catalog counter show object object_name`

See the man pages for more information about which objects and counters are available.

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

Examples

The following command displays descriptions of selected statistic objects related to CIFS and SMB access in the cluster as seen at the advanced privilege level:

```
cluster1::> set -privilege advanced

Warning: These advanced commands are potentially dangerous; use them only
when directed to do so by support personnel.
Do you want to continue? {y|n}: y

cluster1::*> statistics catalog object show -object audit
    audit_ng                    CM object for exporting audit_ng
performance counters

cluster1::*> statistics catalog object show -object cifs
    cifs                        The CIFS object reports activity of the
                                Common Internet File System protocol
                                ...

cluster1::*> statistics catalog object show -object nblade_cifs
    nblade_cifs                 The Common Internet File System (CIFS)
                                protocol is an implementation of the
Server
                                ...

cluster1::*> statistics catalog object show -object smb1
    smb1                        These counters report activity from the
SMB
                                revision of the protocol. For information
                                ...

cluster1::*> statistics catalog object show -object smb2
    smb2                        These counters report activity from the
                                SMB2/SMB3 revision of the protocol. For
                                ...

cluster1::*> statistics catalog object show -object hashd
    hashd                       The hashd object provides counters to
measure
                                the performance of the BranchCache hash
daemon.
cluster1::*> set -privilege admin
```

The following command displays information about some of the counters for the `cifs` object as seen at the advanced privilege level:

> This example does not display all of the available counters for the `cifs` object; output is truncated.

```
cluster1::> set -privilege advanced

Warning: These advanced commands are potentially dangerous; use them only
when directed to do so by support personnel.
Do you want to continue? {y|n}: y

cluster1::*> statistics catalog counter show -object cifs

Object: cifs
    Counter                     Description
    --------------------------
-----------------------------------------------
    active_searches             Number of active searches over SMB and
SMB2
    auth_reject_too_many        Authentication refused after too many
                                requests were made in rapid succession
    avg_directory_depth         Average number of directories crossed by
SMB
                                and SMB2 path-based commands
    ...                         ...

cluster2::> statistics start -object client -sample-id
Object: client
    Counter                                                     Value
    -------------------------------- --------------------------------
    cifs_ops                                                        0
    cifs_read_ops                                                   0
    cifs_read_recv_ops                                              0
    cifs_read_recv_size                                            0B
    cifs_read_size                                                 0B
    cifs_write_ops                                                  0
    cifs_write_recv_ops                                             0
    cifs_write_recv_size                                           0B
    cifs_write_size                                                0B
    instance_name                                vserver_1:10.72.205.179
    instance_uuid                                       2:10.72.205.179
    local_ops                                                       0
    mount_ops                                                       0


[...]
```

Related information

Displaying statistics

Display statistics

You can display various statistics, including statistics about CIFS and SMB, auditing, and BranchCache hashes, to monitor performance and diagnose issues.

Before you begin

You must have collected data samples by using the `statistics start` and `statistics stop` commands before you can display information about objects.

Steps

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

If you want to display statistics for…	Enter…
All versions of SMB	`statistics show -object cifs`
SMB 1.0	`statistics show -object smb1`
SMB 2.x and SMB 3.0	`statistics show -object smb2`
CIFS subsystem of the node	`statistics show -object nblade_cifs`
Multiprotocol audit	`statistics show -object audit_ng`
BranchCache hash service	`statistics show -object hashd`
Dynamic DNS	`statistics show -object ddns_update`

   See the man page for each command for more information.

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

Related information

Determining which statistics objects and counters are available

Monitoring SMB signed session statistics

Displaying BranchCache statistics

Using statistics to monitor automatic node referral activity

SMB configuration for Microsoft Hyper-V and SQL Server

Performance monitoring setup

Copyright information

Trademark information