



# **Manage file security permissions and audit policies**

ONTAP 9.15.1 REST API reference

NetApp  
September 11, 2024

# Table of Contents

- Manage file security permissions and audit policies ..... 1
  - Protocols file-security permissions svm.uuid path endpoint overview ..... 1
  - Remove all SLAG ACLs for a path ..... 16
  - Retrieve file permissions ..... 19
  - Update the SD information ..... 28
  - Apply an SD to a path ..... 42
  - Add a new SACL or DACL ACE ..... 56
  - Delete a SACL or DACL ACL ..... 66
  - Update SACLs or DACLs ..... 75

# Manage file security permissions and audit policies

## Protocols file-security permissions svm.uuid path endpoint overview

### Overview

Using this API, You can manage NTFS file security and audit policies of file or directory without the need of a client. It works similar to what you could do with a `cacls` in windows client. It will create an NTFS security descriptor(SD) to which you can add access control entries (ACEs) to the discretionary access control list (DACL) and the system access control list (SACL). Generally, an SD contains following information:

- Security identifiers (SIDs) for the owner and primary group of an object. A security identifier (SID) is a unique value of variable length used to identify a trustee. Each account has a unique SID issued by an authority, such as a Windows domain controller, and is stored in a security database.
- A DACL identifies the trustees that are allowed or denied access to a securable object. When a process tries to access a securable object, the system checks the ACEs in the object's DACL to determine whether to grant access to it.
- A SACL enables administrators to log attempts to access a secured object. Each ACE specifies the types of access attempts by a specified trustee that cause the system to generate a record in the security event log. An ACE in a SACL can generate audit records when an access attempt fails, when it succeeds, or both.
- A set of control bits that qualify the meaning of a SD or its individual members.

Currently, in ONTAP CLI, creating and applying NTFS ACLs is a 5-step process:

- Create an SD.
- Add DACLs and SACLs to the NTFS SD. If you want to audit file and directory events, you must configure auditing on the Vserver, in addition, to adding a SACL to the SD.
- Create a file/directory security policy. This step associates the policy with a SVM.
- Create a policy task. A policy task refers to a single operation to apply to a file (or folder) or to a set of files (or folders). Among other things, the task defines which SD to apply to a path.
- Apply a policy to the associated SVM.

This REST API to set the DACL/SACL is similar to the windows GUI. The approach used here has been simplified by combining all steps into a single step. The REST API uses only minimal and mandatory parameters to create access control entries (ACEs), which can be added to the discretionary access control list (DACL) and the system access control list (SACL). Based on information provided, SD is created and applied on the target path.

Beginning with ONTAP 9.10.1, SLAG (Storage-Level Access Guard) ACLs can also be configured through these endpoints. SLAG is designed to be set on a volume or `qtree`. Storage-level security cannot be revoked from a client, not even by a system (Windows or UNIX) administrator. It is designed to be modified by storage administrators only, which precedes the share/export permission and the Windows ACLs or UNIX mode bits. Similar to configuring file-directory ACLs, configuring SLAG ACLs is also simplified by combining all steps into a single step.

## Examples

### Creating a new SD

Use this endpoint to apply a fresh set of SACLs and DACLs. A new SD is created based on the input parameters and it replaces the old SD for the given target path:

```
# The API:
POST /protocols/file-security/permissions/{svm.uuid}/{path}

# The call:
curl -X POST "https://10.140.101.39/api/protocols/file-
security/permissions/9479099d-5b9f-11eb-9c4e-
0050568e8682/%2Fparent?return_timeout=0" -H "accept: application/json" -H
"authorization: Basic YWRtaW46bmV0YXBwMSE=" -H "Content-Type:
application/json" -d "{ \"acls\": [ { \"access\": \"access_allow\",
\"advanced_rights\": { \"append_data\": true, \"delete\": true,
\"delete_child\": true, \"execute_file\": true, \"full_control\": true,
\"read_attr\": true, \"read_data\": true, \"read_ea\": true,
\"read_perm\": true, \"write_attr\": true, \"write_data\": true,
\"write_ea\": true, \"write_owner\": true, \"write_perm\": true },
\"apply_to\": { \"files\": true, \"sub_folders\": true, \"this_folder\":
true }, \"user\": \"administrator\" } ], \"control_flags\": \"32788\",
\"group\": \"S-1-5-21-2233347455-2266964949-1780268902-69700\",
\"ignore_paths\": [ \"/parent/child2\" ], \"owner\": \"S-1-5-21-
2233347455-2266964949-1780268902-69304\", \"propagation_mode\":
\"propagate\"}"

# The response:
{
  "job": {
    "uuid": "3015c294-5bbc-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/3015c294-5bbc-11eb-9c4e-0050568e8682"
      }
    }
  }
}
```

### Configuring a new set of SLAG DACLs and SACLs

Use this endpoint to apply a fresh set of SLAG DACLs and SACLs. A new SD is created based on the input

parameters and it replaces the old SLAG permissions for the given target path:

```
# The API:
POST /protocols/file-security/permissions/{svm.uuid}/{path}

# The call:
curl -X POST "https://<mgmt-ip>/api/protocols/file-
security/permissions/9f738ac5-c502-11eb-b82c-
0050568e5902/%2Ftest_vol?return_timeout=0" -H "accept: application/json"
-H "Content-Type: application/json" -d "{ \"access_control\": \"slag\",
\"acls\": [ { \"access\": \"access_allow\",
\"advanced_rights\": { \"append_data\": true, \"delete\":
true, \"delete_child\": true, \"execute_file\": true,
\"full_control\": true, \"read_attr\": true, \"read_data\":
true, \"read_ea\": true, \"read_perm\": true,
\"write_attr\": true, \"write_data\": true, \"write_ea\":
true, \"write_owner\": true, \"write_perm\": true },
\"apply_to\": { \"files\": true, \"sub_folders\": true,
\"this_folder\": true }, \"user\": \"user1\" },{
\"access\": \"audit_success\", \"advanced_rights\": {
\"append_data\": true, \"delete\": true, \"delete_child\":
true, \"execute_file\": true, \"full_control\": true,
\"read_attr\": true, \"read_data\": true, \"read_ea\": true,
\"read_perm\": true, \"write_attr\": true, \"write_data\":
true, \"write_ea\": true, \"write_owner\": true,
\"write_perm\": true }, \"apply_to\": { \"files\": true,
\"sub_folders\": true, \"this_folder\": true }, \"user\":
\"user2\" } ]}"

# The response:
{
  "job": {
    "uuid": "9938d743-d566-11eb-ad60-0050568e5902",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/9938d743-d566-11eb-ad60-0050568e5902"
      }
    }
  }
}
```

## Retrieving file permissions

Use this endpoint to retrieve all the security and auditing information of a directory or file:

```
# The API:
GET /protocols/file-security/permissions/{svm.uuid}/{path}

# The call:
curl -X GET "https://10.140.101.39/api/protocols/file-
security/permissions/9479099d-5b9f-11eb-9c4e-0050568e8682/%2Fparent" -H
"accept: application/json" -H "authorization: Basic YWRtaW46bmV0YXBwMSE="

# The response:
{
  "svm": {
    "uuid": "9479099d-5b9f-11eb-9c4e-0050568e8682",
    "name": "vs1"
  },
  "path": "/parent",
  "owner": "BUILTIN\Administrators",
  "group": "BUILTIN\Administrators",
  "control_flags": "0x8014",
  "acls": [
    {
      "user": "BUILTIN\Administrators",
      "access": "access_allow",
      "apply_to": {
        "files": true,
        "sub_folders": true,
        "this_folder": true
      },
      "advanced_rights": {
        "append_data": true,
        "delete": true,
        "delete_child": true,
        "execute_file": true,
        "full_control": true,
        "read_attr": true,
        "read_data": true,
        "read_ea": true,
        "read_perm": true,
        "write_attr": true,
        "write_data": true,
        "write_ea": true,
        "write_owner": true,

```

```

    "synchronize": true,
    "write_perm": true
  },
  "access_control": "file_directory"
},
{
  "user": "BUILTIN\\Users",
  "access": "access_allow",
  "apply_to": {
    "files": true,
    "sub_folders": true,
    "this_folder": true
  },
  "advanced_rights": {
    "append_data": true,
    "delete": true,
    "delete_child": true,
    "execute_file": true,
    "full_control": true,
    "read_attr": true,
    "read_data": true,
    "read_ea": true,
    "read_perm": true,
    "write_attr": true,
    "write_data": true,
    "write_ea": true,
    "write_owner": true,
    "synchronize": true,
    "write_perm": true
  },
  "access_control": "file_directory"
},
{
  "user": "CREATOR OWNER",
  "access": "access_allow",
  "apply_to": {
    "files": true,
    "sub_folders": true,
    "this_folder": true
  },
  "advanced_rights": {
    "append_data": true,
    "delete": true,
    "delete_child": true,
    "execute_file": true,
    "full_control": true,

```

```

    "read_attr": true,
    "read_data": true,
    "read_ea": true,
    "read_perm": true,
    "write_attr": true,
    "write_data": true,
    "write_ea": true,
    "write_owner": true,
    "synchronize": true,
    "write_perm": true
  },
  "access_control": "file_directory"
},
{
  "user": "Everyone",
  "access": "access_allow",
  "apply_to": {
    "files": true,
    "sub_folders": true,
    "this_folder": true
  },
  "advanced_rights": {
    "append_data": true,
    "delete": true,
    "delete_child": true,
    "execute_file": true,
    "full_control": true,
    "read_attr": true,
    "read_data": true,
    "read_ea": true,
    "read_perm": true,
    "write_attr": true,
    "write_data": true,
    "write_ea": true,
    "write_owner": true,
    "synchronize": true,
    "write_perm": true
  },
  "access_control": "file_directory"
},
{
  "user": "NT AUTHORITY\\SYSTEM",
  "access": "access_allow",
  "apply_to": {
    "files": true,
    "sub_folders": true,

```



```
    "this_folder": true
  },
  "advanced_rights": {
    "append_data": true,
    "delete": true,
    "delete_child": true,
    "execute_file": true,
    "full_control": true,
    "read_attr": true,
    "read_data": true,
    "read_ea": true,
    "read_perm": true,
    "write_attr": true,
    "write_data": true,
    "write_ea": true,
    "write_owner": true,
    "synchronize": true,
    "write_perm": true
  },
  "access_control": "file_directory"
},
{
  "user": "user1",
  "access": "access_allow",
  "apply_to": {
    "sub_folders": true,
    "this_folder": true
  },
  "advanced_rights": {
    "append_data": true,
    "delete": true,
    "delete_child": true,
    "execute_file": true,
    "full_control": true,
    "read_attr": true,
    "read_data": true,
    "read_ea": true,
    "read_perm": true,
    "write_attr": true,
    "write_data": true,
    "write_ea": true,
    "write_owner": true,
    "synchronize": true,
    "write_perm": true
  },
  "access_control": "slag"
```

```

},
{
  "user": "user1",
  "access": "access_allow",
  "apply_to": {
    "files": true,
  },
  "advanced_rights": {
    "append_data": true,
    "delete": true,
    "delete_child": true,
    "execute_file": true,
    "full_control": true,
    "read_attr": true,
    "read_data": true,
    "read_ea": true,
    "read_perm": true,
    "write_attr": true,
    "write_data": true,
    "write_ea": true,
    "write_owner": true,
    "synchronize": true,
    "write_perm": true
  },
  "access_control": "slag"
},
{
  "user": "user2",
  "access": "audit_success",
  "apply_to": {
    "sub_folders": true,
    "this_folder": true
  },
  "advanced_rights": {
    "append_data": true,
    "delete": true,
    "delete_child": true,
    "execute_file": true,
    "full_control": true,
    "read_attr": true,
    "read_data": true,
    "read_ea": true,
    "read_perm": true,
    "write_attr": true,
    "write_data": true,
    "write_ea": true,

```

```
    "write_owner": true,  
    "synchronize": true,  
    "write_perm": true  
  },  
  "access_control": "slag"  
},  
{  
  "user": "user2",  
  "access": "audit_success",  
  "apply_to": {  
    "files": true,  
  },  
  "advanced_rights": {  
    "append_data": true,  
    "delete": true,  
    "delete_child": true,  
    "execute_file": true,  
    "full_control": true,  
    "read_attr": true,  
    "read_data": true,  
    "read_ea": true,  
    "read_perm": true,  
    "write_attr": true,  
    "write_data": true,  
    "write_ea": true,  
    "write_owner": true,  
    "synchronize": true,  
    "write_perm": true  
  },  
  "access_control": "slag"  
}  
],  
"inode": 64,  
"security_style": "mixed",  
"effective_style": "ntfs",  
"dos_attributes": "10",  
"text_dos_attr": "----D---",  
"user_id": "0",  
"group_id": "0",  
"mode_bits": 777,  
"text_mode_bits": "rwxrwxrwx"  
}
```

## Updating SD-specific information

Use this end point to update the following information:

- Primary owner of the file/directory.
- Primary group of the file/directory.
- Control flags associated with with SD of the file/directory.

```
# The API:
PATCH /protocols/file-security/permissions/{svm.uuid}/{path}

# The call:
curl -X PATCH "https://10.140.101.39/api/protocols/file-
security/permissions/9479099d-5b9f-11eb-9c4e-
0050568e8682/%2Fparent?return_timeout=0" -H "accept: application/json" -H
"authorization: Basic YWRtaW46bmV0YXBwMSE=" -H "Content-Type:
application/json" -d '{"control_flags\": \"32788\", \"group\":
\"everyone\", \"owner\": \"user1\"}'

# The Response:
{
  "job": {
    "uuid": "6f89e612-5bbd-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/6f89e612-5bbd-11eb-9c4e-0050568e8682"
      }
    }
  }
}
```

## Removing all SLAG ACLs

Use this end point to remove all SLAG ACLs.

```
# The API:
DELETE /protocols/file-security/permissions/{svm.uuid}/{path}

# The call:
curl -X DELETE "https://<mgmt-ip>/api/protocols/file-
security/permissions/713f569f-d4bc-11eb-b24a-
005056ac6ce1/%2Ftest_vol?access_control=slag"
```

## Adding a single file-directory DACL/SACL ACE

Use this endpoint to add a single SACL/DACL ACE for a new user or for an existing user with a different access type (allow or deny). The given ACE is merged with an existing SACL/DACL and based on the type of “propagation-mode”, it is reflected to the child object:

```
# The API:
POST /protocols/file-security/permissions/{svm.uuid}/{path}/acl

# The call:
curl -X POST "https://10.140.101.39/api/protocols/file-
security/permissions/9479099d-5b9f-11eb-9c4e-
0050568e8682/%2Fparent/acl?return_timeout=0&return_records=false" -H
"accept: application/json" -H "authorization: Basic YWRtaW46bmV0YXBwMSE="
-H "Content-Type: application/json" -d "{ \"access\": \"access_allow\",
\"apply_to\": { \"files\": true, \"sub_folders\": true, \"this_folder\":
true }, \"ignore_paths\": [ \"/parent/child2\" ], \"propagation_mode\":
\"propagate\", \"rights\": \"read\", \"user\": \"himanshu\"}"

# The Response:
{
  "job": {
    "uuid": "26185a2f-5bbe-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/26185a2f-5bbe-11eb-9c4e-0050568e8682"
      }
    }
  }
}
```

## Adding a single SLAG DACL/SACL ACE

Use this endpoint to add a single SLAG SACL/DACL ACE to an existing set of ACLs for a user or for an existing user with a different access type (allow or deny).

```
# The API:
POST /protocols/file-security/permissions/{svm.uuid}/{path}/acl

# The call:
curl -X POST "https://<mgmt-ip>/api/protocols/file-
security/permissions/713f569f-d4bc-11eb-b24a-
005056ac6ce1/%2Ftest_vol/acl?return_timeout=0&return_records=false" -H
"accept: application/json" -H "authorization: Basic YWRtaW46bmV0YXBwMSE="
-H "Content-Type: application/json" -d "{ \"access\": \"access_allow\",
\"access_control\": \"slag\", \"advanced_rights\": { \"append_data\":
true, \"delete\": true, \"delete_child\": true, \"execute_file\":
true, \"full_control\": true, \"read_attr\": true, \"read_data\":
true, \"read_ea\": true, \"read_perm\": true, \"write_attr\":
true, \"write_data\": true, \"write_ea\": true, \"write_owner\":
true, \"write_perm\": true }, \"apply_to\": { \"files\": true,
\"sub_folders\": true, \"this_folder\": true }, \"user\": \"user1\"}"

# The Response:
{
  "job": {
    "uuid": "7fa5f53f-d570-11eb-b24a-005056ac6ce1",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/7fa5f53f-d570-11eb-b24a-005056ac6ce1"
      }
    }
  }
}
```

## Updating existing SACL/DACL ACE

Use this endpoint to update the rights/advanced rights for an existing user, for a specified path. You cannot update the access type using this end point. Based on the type of “propagation-mode”, it is reflected to the child object:

```

# The API:
PATCH /protocols/file-security/permissions/{svm.uuid}/{path}/acl/{user}
The Call:
curl -X PATCH "https://10.140.101.39/api/protocols/file-
security/permissions/9479099d-5b9f-11eb-9c4e-
0050568e8682/%2Fparent/acl/himanshu?return_timeout=0" -H "accept:
application/json" -H "authorization: Basic YWRtaW46bmV0YXBwMSE=" -H
"Content-Type: application/json" -d "{ \"access\": \"access_allow\",
\"advanced_rights\": { \"append_data\": true, \"delete\": true,
\"delete_child\": true, \"execute_file\": true, \"full_control\": true,
\"read_attr\": false, \"read_data\": false, \"read_ea\": false,
\"read_perm\": false, \"write_attr\": true, \"write_data\": true,
\"write_ea\": true, \"write_owner\": true, \"write_perm\": true },
\"apply_to\": { \"files\": true, \"sub_folders\": true, \"this_folder\":
true }, \"ignore_paths\": [ \"/parent/child2\" ], \"propagation_mode\":
\"propagate\"}"
The Response:
{
  "job": {
    "uuid": "72067401-5bbf-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/72067401-5bbf-11eb-9c4e-0050568e8682"
      }
    }
  }
}

```

---

### Updating an existing SLG SACL/DACL ACE

Use this endpoint to update the SLAG rights/advanced rights for an existing user, for a specified path. You cannot update the access type using this end point.

---

```
# The API:
PATCH /protocols/file-security/permissions/{svm.uuid}/{path}/acl/{user}
The Call:
curl -X PATCH "https://<mgmt-ip>/api/protocols/file-
security/permissions/713f569f-d4bc-11eb-b24a-
005056ac6ce1/%2Ftest_vol/acl/user1?return_records=false&return_timeout=0"
-H "accept: application/json" -H "authorization: Basic
YWRtaW46bmV0YXBwMSE=" -H "Content-Type: application/json" -d "{
\"access\": \"access_allow\", \"access_control\": \"slag\",
\"apply_to\": { \"files\": true, \"sub_folders\": true,
\"this_folder\": true }, \"rights\": \"read\"}"
The Response:
{
  "job": {
    "uuid": "3d21abcd-d571-11eb-b24a-005056ac6ce1",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/3d21abcd-d571-11eb-b24a-005056ac6ce1"
      }
    }
  }
}
```

---

### Deleting an existing SACL/DACL ACE

Use this endpoint to delete any of the existing rights/advanced\_rights for a user. Based on the type of "propagation-mode", it is reflected to the child object:

---



```
# The API:
DELETE /protocols/file-security/permissions/{svm.uuid}/{path}/acl/{user}

# The call:
curl -X DELETE "https://10.140.101.39/api/protocols/file-
security/permissions/9479099d-5b9f-11eb-9c4e-
0050568e8682/%2Fparent/acl/himanshu?return_timeout=0" -H "accept:
application/json" -H "authorization: Basic YWRtaW46bmV0YXBwMSE=" -H
"Content-Type: application/json" -d "{ \"access\": \"access_allow\",
\"apply_to\": { \"files\": true, \"sub_folders\": true, \"this_folder\":
true }, \"ignore_paths\": [ \"/parent/child2\" ], \"propagation_mode\":
\"propagate\"}"

# The response:
{
  "job": {
    "uuid": "e5683b61-5bbf-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/e5683b61-5bbf-11eb-9c4e-0050568e8682"
      }
    }
  }
}
```

---

### Deleting an existing SLAG SACL/DACL ACE

Use this endpoint to delete any SLAG ACE for a user.

---

```

# The API:
DELETE /protocols/file-security/permissions/{svm.uuid}/{path}/acl/{user}

# The call:
curl -X DELETE "https://<mgmt-ip>/api/protocols/file-
security/permissions/713f569f-d4bc-11eb-b24a-
005056ac6ce1/%2Ftest_vol/acl/user1?return_records=false&return_timeout=0"
-H "accept: application/json" -H "authorization: Basic
YWRtaW46bmV0YXBwMSE=" -H "Content-Type: application/json" -d "{
\"access\": \"access_allow\", \"access_control\": \"slag\",
\"apply_to\": { \"files\": true, \"sub_folders\": true,
\"this_folder\": true }}"

# The response:
{
  "job": {
    "uuid": "10c29534-d572-11eb-b24a-005056ac6ce1",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/10c29534-d572-11eb-b24a-005056ac6ce1"
      }
    }
  }
}

```

## Remove all SLAG ACLs for a path

```
DELETE /protocols/file-security/permissions/{svm.uuid}/{path}
```

**Introduced In:** 9.10

Remove all SLAG ACLs for specified path. Bulk deletion is supported only for SLAG You must keep the following points in mind while using these endpoints:

- Do not pass additional arguments that are not required.

### Related ONTAP Commands

- `vserver security file-directory remove-slag`

### Parameters

| Name           | Type   | In    | Required | Description  |
|----------------|--------|-------|----------|--|
| path           | string | path  | True     | target path  |
| access_control | string | query | False    | Remove all SLAG ACLs. Currently bulk deletion of file-directory ACLs is not supported.<br><br>• enum: ["slag"] |
| svm.uuid       | string | path  | True     | UUID of the SVM to which this object belongs.  |

## Response

Status: 200, Ok

## Error

Status: Default

### ONTAP Error Response Codes

| Error Code | Description                                     |
|------------|---|
| 655865     | The specified file or directory does not exist. |
| 10485811   | Access is a required field.                     |
| 1260882    | Specified SVM not found.                        |
| 6691623    | User is not authorized.                         |

| Name  | Type                           | Description |
|-------|--------------------------------|-------------|
| error | <a href="#">returned_error</a> |             |

## Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

## Definitions

### See Definitions

#### error\_arguments

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

#### returned\_error

| Name      | Type                                     | Description                                 |
|-----------|--|---|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments                           |
| code      | string                                   | Error code                                  |
| message   | string                                   | Error message                               |
| target    | string                                   | The target parameter that caused the error. |

# Retrieve file permissions

GET /protocols/file-security/permissions/{svm.uuid}/{path}

Introduced In: 9.9

Retrieves file permissions

## Related ONTAP commands

- `vserver security file-directory show`

## Parameters

| Name     | Type          | In    | Required | Description                                   |
|----------|---------------|-------|----------|---|
| path     | string        | path  | True     | target path                                   |
| fields   | array[string] | query | False    | Specify the fields to return.                 |
| svm.uuid | string        | path  | True     | UUID of the SVM to which this object belongs. |

## Response

Status: 200, Ok

| Name           | Type   | Description   |
|----------------|--------|---|
| access_control | string | An Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable). |

| Name            | Type                         | Description  |
|-----------------|------------------------------|--|
| acls            | array[ <a href="#">acl</a> ] | A discretionary access security list (DACL) identifies the trustees that are allowed or denied access to a securable object. When a process tries to access a securable object, the system checks the access control entries (ACEs) in the object's DACL to determine whether to grant access to it. |
| control_flags   | string                       | Specifies the control flags in the SD. It is a Hexadecimal Value.  |
| dos_attributes  | string                       | Specifies the file attributes on this file or directory.   |
| effective_style | string                       | Specifies the effective style of the SD. The following values are supported: <ul style="list-style-type: none"> <li>• unix - UNIX style</li> <li>• ntfs - NTFS style</li> <li>• mixed - Mixed style</li> <li>• unified - Unified style</li> </ul>  |
| group           | string                       | Specifies the owner's primary group. You can specify the owner group using either a group name or SID.   |
| group_id        | string                       | Specifies group ID on this file or directory.  |
| ignore_paths    | array[string]                | Specifies that permissions on this file or directory cannot be replaced.   |
| inode           | integer                      | Specifies the File Inode number.   |
| mode_bits       | integer                      | Specifies the mode bits on this file or directory.   |

| Name             | Type   | Description  |
|------------------|--------|--|
| owner            | string | Specifies the owner of the SD. You can specify the owner using either a user name or security identifier (SID). The owner of the SD can modify the permissions on the file (or folder) or files (or folders) to which the SD is applied and can give other users the right to take ownership of the object or objects to which the SD is applied.  |
| propagation_mode | string | <p>Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are:</p> <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• ignore - ignore inheritable permissions</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |
| security_style   | string | <p>Specifies the security style of the SD. The following values are supported:</p> <ul style="list-style-type: none"> <li>• unix - UNIX style</li> <li>• ntfs - NTFS style</li> <li>• mixed - Mixed style</li> <li>• unified - Unified style</li> </ul>  |
| text_dos_attr    | string | Specifies the textual format of file attributes on this file or directory.   |
| text_mode_bits   | string | Specifies the textual format of mode bits on this file or directory.   |
| user_id          | string | Specifies user ID of this file or directory.   |

## Example response

```
{
  "access_control": "file_directory",
  "acls": [
    {
      "access": "access_allow",
      "access_control": "file_directory",
      "inherited": 1,
      "rights": "full_control",
      "user": "S-1-5-21-2233347455-2266964949-1780268902-69304"
    }
  ],
  "control_flags": "8014",
  "dos_attributes": "10",
  "effective_style": "mixed",
  "group": "S-1-5-21-2233347455-2266964949-1780268902-69700",
  "group_id": "2",
  "ignore_paths": [
    "/dir1/dir2/",
    "/parent/dir3"
  ],
  "inode": 64,
  "mode_bits": 777,
  "owner": "S-1-5-21-2233347455-2266964949-1780268902-69304",
  "propagation_mode": "propagate",
  "security_style": "ntfs",
  "text_dos_attr": "---A---",
  "text_mode_bits": "rwxrwxrwx",
  "user_id": "10"
}
```

## Error

Status: Default

### ONTAP Error Response Codes

| Error Code | Description                                     |
|------------|---|
| 655865     | The specified file or directory does not exist. |
| 1260882    | Specified SVM not found.                        |
| 6691623    | User is not authorized.                         |



| Error Code | Description   |
|------------|---|
| 4849676    | The specified Windows user or group does not exist. |

| Name  | Type                           | Description |
|-------|--------------------------------|-------------|
| error | <a href="#">returned_error</a> |             |

### Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

### Definitions

## See Definitions

### advanced\_rights

Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.

| Name         | Type    | Description               |
|--------------|---------|---------------------------|
| append_data  | boolean | Append DAta               |
| delete       | boolean | Delete                    |
| delete_child | boolean | Delete Child              |
| execute_file | boolean | Execute File              |
| full_control | boolean | Full Control              |
| read_attr    | boolean | Read Attributes           |
| read_data    | boolean | Read Data                 |
| read_ea      | boolean | Read Extended Attributes  |
| read_perm    | boolean | Read Permissions          |
| synchronize  | boolean | Synchronize               |
| write_attr   | boolean | Write Attributes          |
| write_data   | boolean | Write Data                |
| write_ea     | boolean | Write Extended Attributes |
| write_owner  | boolean | Write Owner               |
| write_perm   | boolean | Write Permission          |

### apply\_to

Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.

| Name  | Type    | Description    |
|-------|---------|----------------|
| files | boolean | Apply to Files |

| <b>Name</b> | <b>Type</b> | <b>Description</b>        |
|-------------|-------------|---------------------------|
| sub_folders | boolean     | Apply to all sub-folders  |
| this_folder | boolean     | Apply only to this folder |

acl

An ACE is an element in an access control list (ACL). An ACL can have zero or more ACEs. Each ACE controls or monitors access to an object by a specified trustee.

| Name   | Type   | Description   |
|--------|--------|---|
| access | string | <p>Specifies whether the ACL is for DACL or SACL. The available values are:</p> <ul style="list-style-type: none"> <li>• access_allow - DACL for allow access</li> <li>• access_deny - DACL for deny access</li> <li>• access_allowed_callback - CALLBACK for allowed access</li> <li>• access_denied_callback - CALLBACK for denied access</li> <li>• access_allowed_callback_object - CALLBACK OBJECT for allowed access</li> <li>• access_denied_callback_object - CALLBACK OBJECT for denied access</li> <li>• system_audit_callback - SYSTEM Audit Callback ace</li> <li>• system_audit_callback_object - SYSTEM Audit Callback Object ace</li> <li>• system_resource_attribute - SYSTEM Resource Attribute</li> <li>• system_scoped_policy_id - SYSTEM Scope Policy ID</li> <li>• audit_success - SACL for success access</li> <li>• audit_failure - SACL for failure access</li> <li>• audit_success_and_failure - SACL for both success and failure access</li> </ul> |

| Name            | Type            | Description   |
|-----------------|-----------------|---|
| access_control  | string          | An Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable). |
| advanced_rights | advanced_rights | Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.   |
| apply_to        | apply_to        | Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.   |
| inherited       | boolean         | Indicates whether or not the ACE flag is inherited.   |
| rights          | string          | Specifies the access rights controlled by the ACE for the account specified. The "rights" parameter is mutually exclusive with the "advanced_rights" parameter. If you specify the "rights" parameter, you can specify one of the following "rights" values ("- " or " _ " is accepted as the delimiter).   |
| user            | string          | Specifies the account to which the ACE applies. You can specify either name or SID.   |

error\_arguments

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

returned\_error

| Name      | Type                                     | Description                                 |
|-----------|--|---|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments                           |
| code      | string                                   | Error code                                  |
| message   | string                                   | Error message                               |
| target    | string                                   | The target parameter that caused the error. |

## Update the SD information

PATCH /protocols/file-security/permissions/{svm.uuid}/{path}

**Introduced In:** 9.9

Updates SD specific Information. For example, owner, group and control-flags. SD specific information of SLAG ACLs is not modifiable.

### Related ONTAP commands

- `vserver security file-directory ntfs modify`

### Parameters

| Name | Type   | In   | Required | Description |
|------|--------|------|----------|-------------|
| path | string | path | True     | target path |

| Name           | Type    | In    | Required | Description  |
|----------------|---------|-------|----------|--|
| return_timeout | integer | query | False    | <p>The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.</p> <ul style="list-style-type: none"> <li>• Default value: 1</li> <li>• Max value: 120</li> <li>• Min value: 0</li> </ul> |
| svm.uuid       | string  | path  | True     | <p>UUID of the SVM to which this object belongs.</p>   |

## Request Body

| Name            | Type                        | Description   |
|-----------------|-----------------------------|---|
| access_control  | string                      | An Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable). |
| acls            | array <a href="#">[acl]</a> | A discretionary access security list (DACL) identifies the trustees that are allowed or denied access to a securable object. When a process tries to access a securable object, the system checks the access control entries (ACEs) in the object's DACL to determine whether to grant access to it.  |
| control_flags   | string                      | Specifies the control flags in the SD. It is a Hexadecimal Value.   |
| dos_attributes  | string                      | Specifies the file attributes on this file or directory.  |
| effective_style | string                      | Specifies the effective style of the SD. The following values are supported: <ul style="list-style-type: none"> <li>• unix - UNIX style</li> <li>• ntfs - NTFS style</li> <li>• mixed - Mixed style</li> <li>• unified - Unified style</li> </ul>   |
| group           | string                      | Specifies the owner's primary group. You can specify the owner group using either a group name or SID.  |
| group_id        | string                      | Specifies group ID on this file or directory.   |



| Name             | Type          | Description  |
|------------------|---------------|--|
| ignore_paths     | array[string] | Specifies that permissions on this file or directory cannot be replaced.   |
| inode            | integer       | Specifies the File Inode number.   |
| mode_bits        | integer       | Specifies the mode bits on this file or directory.   |
| owner            | string        | Specifies the owner of the SD. You can specify the owner using either a user name or security identifier (SID). The owner of the SD can modify the permissions on the file (or folder) or files (or folders) to which the SD is applied and can give other users the right to take ownership of the object or objects to which the SD is applied.  |
| propagation_mode | string        | <p>Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are:</p> <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• ignore - ignore inheritable permissions</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |
| security_style   | string        | <p>Specifies the security style of the SD. The following values are supported:</p> <ul style="list-style-type: none"> <li>• unix - UNIX style</li> <li>• ntfs - NTFS style</li> <li>• mixed - Mixed style</li> <li>• unified - Unified style</li> </ul>  |

| Name           | Type   | Description  |
|----------------|--------|--|
| text_dos_attr  | string | Specifies the textual format of file attributes on this file or directory. |
| text_mode_bits | string | Specifies the textual format of mode bits on this file or directory.       |
| user_id        | string | Specifies user ID of this file or directory.                               |

### Example request

```
{
  "access_control": "file_directory",
  "acls": [
    {
      "access": "access_allow",
      "access_control": "file_directory",
      "inherited": 1,
      "rights": "full_control",
      "user": "S-1-5-21-2233347455-2266964949-1780268902-69304"
    }
  ],
  "control_flags": "8014",
  "dos_attributes": "10",
  "effective_style": "mixed",
  "group": "S-1-5-21-2233347455-2266964949-1780268902-69700",
  "group_id": "2",
  "ignore_paths": [
    "/dir1/dir2/",
    "/parent/dir3"
  ],
  "inode": 64,
  "mode_bits": 777,
  "owner": "S-1-5-21-2233347455-2266964949-1780268902-69304",
  "propagation_mode": "propagate",
  "security_style": "ntfs",
  "text_dos_attr": "---A---",
  "text_mode_bits": "rwxrwxrwx",
  "user_id": "10"
}
```

## Response

Status: 200, Ok

| Name | Type     | Description |
|------|----------|-------------|
| job  | job_link |             |

## Example response

```
{
  "job": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "string"
  }
}
```

## Response

Status: 202, Accepted

## Error

Status: Default

## ONTAP Error Response Codes

| Error Code | Description  |
|------------|--|
| 655865     | The specified file or directory does not exist.        |
| 1260882    | Specified SVM not found.                               |
| 6691623    | User is not authorized.                                |
| 4849676    | The specified Windows user or group does not exist.    |
| 10485814   | The value provided for field control_flags is invalid. |

| Name  | Type                           | Description |
|-------|--------------------------------|-------------|
| error | <a href="#">returned_error</a> |             |

### Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

### Definitions

## See Definitions

### advanced\_rights

Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.

| Name         | Type    | Description               |
|--------------|---------|---------------------------|
| append_data  | boolean | Append DAta               |
| delete       | boolean | Delete                    |
| delete_child | boolean | Delete Child              |
| execute_file | boolean | Execute File              |
| full_control | boolean | Full Control              |
| read_attr    | boolean | Read Attributes           |
| read_data    | boolean | Read Data                 |
| read_ea      | boolean | Read Extended Attributes  |
| read_perm    | boolean | Read Permissions          |
| synchronize  | boolean | Synchronize               |
| write_attr   | boolean | Write Attributes          |
| write_data   | boolean | Write Data                |
| write_ea     | boolean | Write Extended Attributes |
| write_owner  | boolean | Write Owner               |
| write_perm   | boolean | Write Permission          |

### apply\_to

Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.

| Name  | Type    | Description    |
|-------|---------|----------------|
| files | boolean | Apply to Files |

| <b>Name</b> | <b>Type</b> | <b>Description</b>        |
|-------------|-------------|---------------------------|
| sub_folders | boolean     | Apply to all sub-folders  |
| this_folder | boolean     | Apply only to this folder |

## acl

An ACE is an element in an access control list (ACL). An ACL can have zero or more ACEs. Each ACE controls or monitors access to an object by a specified trustee.

| Name   | Type   | Description   |
|--------|--------|---|
| access | string | <p>Specifies whether the ACL is for DACL or SACL. The available values are:</p> <ul style="list-style-type: none"> <li>• access_allow - DACL for allow access</li> <li>• access_deny - DACL for deny access</li> <li>• access_allowed_callback - CALLBACK for allowed access</li> <li>• access_denied_callback - CALLBACK for denied access</li> <li>• access_allowed_callback_object - CALLBACK OBJECT for allowed access</li> <li>• access_denied_callback_object - CALLBACK OBJECT for denied access</li> <li>• system_audit_callback - SYSTEM Audit Callback ace</li> <li>• system_audit_callback_object - SYSTEM Audit Callback Object ace</li> <li>• system_resource_attribute - SYSTEM Resource Attribute</li> <li>• system_scoped_policy_id - SYSTEM Scope Policy ID</li> <li>• audit_success - SACL for success access</li> <li>• audit_failure - SACL for failure access</li> <li>• audit_success_and_failure - SACL for both success and failure access</li> </ul> |

| Name            | Type            | Description   |
|-----------------|-----------------|---|
| access_control  | string          | An Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable). |
| advanced_rights | advanced_rights | Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.   |
| apply_to        | apply_to        | Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.   |
| inherited       | boolean         | Indicates whether or not the ACE flag is inherited.   |
| rights          | string          | Specifies the access rights controlled by the ACE for the account specified. The "rights" parameter is mutually exclusive with the "advanced_rights" parameter. If you specify the "rights" parameter, you can specify one of the following "rights" values ("- " or " _ " is accepted as the delimiter).   |
| user            | string          | Specifies the account to which the ACE applies. You can specify either name or SID.   |

file\_directory\_security

Manages New Technology File System (NTFS) security and NTFS audit policies.



| Name            | Type                        | Description   |
|-----------------|-----------------------------|---|
| access_control  | string                      | An Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable). |
| acls            | array <a href="#">[acl]</a> | A discretionary access security list (DACL) identifies the trustees that are allowed or denied access to a securable object. When a process tries to access a securable object, the system checks the access control entries (ACEs) in the object's DACL to determine whether to grant access to it.  |
| control_flags   | string                      | Specifies the control flags in the SD. It is a Hexadecimal Value.   |
| dos_attributes  | string                      | Specifies the file attributes on this file or directory.  |
| effective_style | string                      | Specifies the effective style of the SD. The following values are supported: <ul style="list-style-type: none"> <li>• unix - UNIX style</li> <li>• ntfs - NTFS style</li> <li>• mixed - Mixed style</li> <li>• unified - Unified style</li> </ul>   |
| group           | string                      | Specifies the owner's primary group. You can specify the owner group using either a group name or SID.  |

| Name             | Type          | Description  |
|------------------|---------------|--|
| group_id         | string        | Specifies group ID on this file or directory.  |
| ignore_paths     | array[string] | Specifies that permissions on this file or directory cannot be replaced.   |
| inode            | integer       | Specifies the File Inode number.   |
| mode_bits        | integer       | Specifies the mode bits on this file or directory.   |
| owner            | string        | Specifies the owner of the SD. You can specify the owner using either a user name or security identifier (SID). The owner of the SD can modify the permissions on the file (or folder) or files (or folders) to which the SD is applied and can give other users the right to take ownership of the object or objects to which the SD is applied.  |
| propagation_mode | string        | <p>Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are:</p> <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• ignore - ignore inheritable permissions</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |

| Name           | Type   | Description  |
|----------------|--------|--|
| security_style | string | Specifies the security style of the SD. The following values are supported: <ul style="list-style-type: none"> <li>• unix - UNIX style</li> <li>• ntfs - NTFS style</li> <li>• mixed - Mixed style</li> <li>• unified - Unified style</li> </ul> |
| text_dos_attr  | string | Specifies the textual format of file attributes on this file or directory.   |
| text_mode_bits | string | Specifies the textual format of mode bits on this file or directory.   |
| user_id        | string | Specifies user ID of this file or directory.   |

href

| Name | Type   | Description |
|------|--------|-------------|
| href | string |             |

\_links

| Name | Type                 | Description |
|------|----------------------|-------------|
| self | <a href="#">href</a> |             |

job\_link

| Name   | Type                   | Description   |
|--------|------------------------|---|
| _links | <a href="#">_links</a> |   |
| uuid   | string                 | The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation. |

error\_arguments

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

returned\_error

| Name      | Type                                     | Description                                 |
|-----------|--|---|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments                           |
| code      | string                                   | Error code                                  |
| message   | string                                   | Error message                               |
| target    | string                                   | The target parameter that caused the error. |

## Apply an SD to a path

POST /protocols/file-security/permissions/{svm.uuid}/{path}

**Introduced In:** 9.9

Applies an SD to the given path. You must keep the following points in mind while using these endpoints:

- Either SLAG ACL/s or file-directory ACL/s can be configured in one API call. Both cannot be configured in the same API call.
- SLAG applies to all files and/or directories in a volume hence, inheritance is not required to be propagated.
- Set `access_control` field to `slag` while configuring SLAG ACLs.
- Set `access_control` field to `file_directory` while configuring file-directory ACLs. By Default `access_control` field is set to `file_directory`.
- For SLAG, valid `apply_to` combinations are "this-folder, sub-folders", "files", "this-folder, sub-folders, files".

### Related ONTAP commands

- `vserver security file-directory ntfs create`
- `vserver security file-directory ntfs dacl add`
- `vserver security file-directory ntfs sacl add`
- `vserver security file-directory policy create`
- `vserver security file-directory policy task add`
- `vserver security file-directory apply`

### Parameters

| Name | Type   | In   | Required | Description |
|------|--------|------|----------|-------------|
| path | string | path | True     | target path |

| Name           | Type    | In    | Required | Description  |
|----------------|---------|-------|----------|--|
| return_timeout | integer | query | False    | <p>The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.</p> <ul style="list-style-type: none"> <li>• Default value: 1</li> <li>• Max value: 120</li> <li>• Min value: 0</li> </ul> |
| svm.uuid       | string  | path  | True     | <p>UUID of the SVM to which this object belongs.</p>   |

## Request Body

| Name            | Type                        | Description   |
|-----------------|-----------------------------|---|
| access_control  | string                      | An Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable). |
| acls            | array <a href="#">[acl]</a> | A discretionary access security list (DACL) identifies the trustees that are allowed or denied access to a securable object. When a process tries to access a securable object, the system checks the access control entries (ACEs) in the object's DACL to determine whether to grant access to it.  |
| control_flags   | string                      | Specifies the control flags in the SD. It is a Hexadecimal Value.   |
| dos_attributes  | string                      | Specifies the file attributes on this file or directory.  |
| effective_style | string                      | Specifies the effective style of the SD. The following values are supported: <ul style="list-style-type: none"> <li>• unix - UNIX style</li> <li>• ntfs - NTFS style</li> <li>• mixed - Mixed style</li> <li>• unified - Unified style</li> </ul>   |
| group           | string                      | Specifies the owner's primary group. You can specify the owner group using either a group name or SID.  |
| group_id        | string                      | Specifies group ID on this file or directory.   |

| Name             | Type          | Description  |
|------------------|---------------|--|
| ignore_paths     | array[string] | Specifies that permissions on this file or directory cannot be replaced.   |
| inode            | integer       | Specifies the File Inode number.   |
| mode_bits        | integer       | Specifies the mode bits on this file or directory.   |
| owner            | string        | Specifies the owner of the SD. You can specify the owner using either a user name or security identifier (SID). The owner of the SD can modify the permissions on the file (or folder) or files (or folders) to which the SD is applied and can give other users the right to take ownership of the object or objects to which the SD is applied.  |
| propagation_mode | string        | <p>Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are:</p> <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• ignore - ignore inheritable permissions</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |
| security_style   | string        | <p>Specifies the security style of the SD. The following values are supported:</p> <ul style="list-style-type: none"> <li>• unix - UNIX style</li> <li>• ntfs - NTFS style</li> <li>• mixed - Mixed style</li> <li>• unified - Unified style</li> </ul>  |

| Name           | Type   | Description  |
|----------------|--------|--|
| text_dos_attr  | string | Specifies the textual format of file attributes on this file or directory. |
| text_mode_bits | string | Specifies the textual format of mode bits on this file or directory.       |
| user_id        | string | Specifies user ID of this file or directory.                               |

### Example request

```
{
  "access_control": "file_directory",
  "acls": [
    {
      "access": "access_allow",
      "access_control": "file_directory",
      "inherited": 1,
      "rights": "full_control",
      "user": "S-1-5-21-2233347455-2266964949-1780268902-69304"
    }
  ],
  "control_flags": "8014",
  "dos_attributes": "10",
  "effective_style": "mixed",
  "group": "S-1-5-21-2233347455-2266964949-1780268902-69700",
  "group_id": "2",
  "ignore_paths": [
    "/dir1/dir2/",
    "/parent/dir3"
  ],
  "inode": 64,
  "mode_bits": 777,
  "owner": "S-1-5-21-2233347455-2266964949-1780268902-69304",
  "propagation_mode": "propagate",
  "security_style": "ntfs",
  "text_dos_attr": "---A---",
  "text_mode_bits": "rwxrwxrwx",
  "user_id": "10"
}
```



## Response

Status: 202, Accepted

| Name | Type     | Description |
|------|----------|-------------|
| job  | job_link |             |

## Example response

```
{
  "job": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "string"
  }
}
```

## Headers

| Name     | Description                               | Type   |
|----------|---|--------|
| Location | Useful for tracking the resource location | string |

## Response

Status: 201, Created

## Error

Status: Default

## ONTAP Error Response Codes

| Error Code | Description   |
|------------|---|
| 655865     | The specified file or directory does not exist.     |
| 10485813   | All values corresponding to rights cannot be false. |

| Error Code | Description  |
|------------|--|
| 10485815   | The field "acls.access_control" is not allowed with POST method.                 |
| 10485810   | User is a required field.  |
| 1260882    | Specified SVM not found.   |
| 6691623    | User is not authorized.  |
| 4849676    | The specified Windows user or group does not exist.                              |
| 4849677    | Failed to convert SID to a Windows name. Reason: "SecD Error: object not found". |
| 10485814   | The value provided for field control_flags is invalid.                           |
| 10485813   | All values cannot be false.  |

| Name  | Type                           | Description |
|-------|--------------------------------|-------------|
| error | <a href="#">returned_error</a> |             |

### Example error

```

{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}

```

### Definitions

## See Definitions

### advanced\_rights

Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.

| Name         | Type    | Description               |
|--------------|---------|---------------------------|
| append_data  | boolean | Append DAta               |
| delete       | boolean | Delete                    |
| delete_child | boolean | Delete Child              |
| execute_file | boolean | Execute File              |
| full_control | boolean | Full Control              |
| read_attr    | boolean | Read Attributes           |
| read_data    | boolean | Read Data                 |
| read_ea      | boolean | Read Extended Attributes  |
| read_perm    | boolean | Read Permissions          |
| synchronize  | boolean | Synchronize               |
| write_attr   | boolean | Write Attributes          |
| write_data   | boolean | Write Data                |
| write_ea     | boolean | Write Extended Attributes |
| write_owner  | boolean | Write Owner               |
| write_perm   | boolean | Write Permission          |

### apply\_to

Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.

| Name  | Type    | Description    |
|-------|---------|----------------|
| files | boolean | Apply to Files |

| <b>Name</b> | <b>Type</b> | <b>Description</b>        |
|-------------|-------------|---------------------------|
| sub_folders | boolean     | Apply to all sub-folders  |
| this_folder | boolean     | Apply only to this folder |

## acl

An ACE is an element in an access control list (ACL). An ACL can have zero or more ACEs. Each ACE controls or monitors access to an object by a specified trustee.

| Name   | Type   | Description   |
|--------|--------|---|
| access | string | <p>Specifies whether the ACL is for DACL or SACL. The available values are:</p> <ul style="list-style-type: none"> <li>• access_allow - DACL for allow access</li> <li>• access_deny - DACL for deny access</li> <li>• access_allowed_callback - CALLBACK for allowed access</li> <li>• access_denied_callback - CALLBACK for denied access</li> <li>• access_allowed_callback_object - CALLBACK OBJECT for allowed access</li> <li>• access_denied_callback_object - CALLBACK OBJECT for denied access</li> <li>• system_audit_callback - SYSTEM Audit Callback ace</li> <li>• system_audit_callback_object - SYSTEM Audit Callback Object ace</li> <li>• system_resource_attribute - SYSTEM Resource Attribute</li> <li>• system_scoped_policy_id - SYSTEM Scope Policy ID</li> <li>• audit_success - SACL for success access</li> <li>• audit_failure - SACL for failure access</li> <li>• audit_success_and_failure - SACL for both success and failure access</li> </ul> |

| Name            | Type                            | Description   |
|-----------------|---------------------------------|---|
| access_control  | string                          | An Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable). |
| advanced_rights | <a href="#">advanced_rights</a> | Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.   |
| apply_to        | <a href="#">apply_to</a>        | Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.   |
| inherited       | boolean                         | Indicates whether or not the ACE flag is inherited.   |
| rights          | string                          | Specifies the access rights controlled by the ACE for the account specified. The "rights" parameter is mutually exclusive with the "advanced_rights" parameter. If you specify the "rights" parameter, you can specify one of the following "rights" values ("- " or " _ " is accepted as the delimiter).   |
| user            | string                          | Specifies the account to which the ACE applies. You can specify either name or SID.   |

file\_directory\_security

Manages New Technology File System (NTFS) security and NTFS audit policies.

| Name            | Type                        | Description   |
|-----------------|-----------------------------|---|
| access_control  | string                      | An Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable). |
| acls            | array <a href="#">[acl]</a> | A discretionary access security list (DACL) identifies the trustees that are allowed or denied access to a securable object. When a process tries to access a securable object, the system checks the access control entries (ACEs) in the object's DACL to determine whether to grant access to it.  |
| control_flags   | string                      | Specifies the control flags in the SD. It is a Hexadecimal Value.   |
| dos_attributes  | string                      | Specifies the file attributes on this file or directory.  |
| effective_style | string                      | Specifies the effective style of the SD. The following values are supported: <ul style="list-style-type: none"> <li>• unix - UNIX style</li> <li>• ntfs - NTFS style</li> <li>• mixed - Mixed style</li> <li>• unified - Unified style</li> </ul>   |
| group           | string                      | Specifies the owner's primary group. You can specify the owner group using either a group name or SID.  |

| Name             | Type          | Description  |
|------------------|---------------|--|
| group_id         | string        | Specifies group ID on this file or directory.  |
| ignore_paths     | array[string] | Specifies that permissions on this file or directory cannot be replaced.   |
| inode            | integer       | Specifies the File Inode number.   |
| mode_bits        | integer       | Specifies the mode bits on this file or directory.   |
| owner            | string        | Specifies the owner of the SD. You can specify the owner using either a user name or security identifier (SID). The owner of the SD can modify the permissions on the file (or folder) or files (or folders) to which the SD is applied and can give other users the right to take ownership of the object or objects to which the SD is applied.  |
| propagation_mode | string        | <p>Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are:</p> <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• ignore - ignore inheritable permissions</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |



| Name           | Type   | Description  |
|----------------|--------|--|
| security_style | string | Specifies the security style of the SD. The following values are supported: <ul style="list-style-type: none"> <li>• unix - UNIX style</li> <li>• ntfs - NTFS style</li> <li>• mixed - Mixed style</li> <li>• unified - Unified style</li> </ul> |
| text_dos_attr  | string | Specifies the textual format of file attributes on this file or directory.   |
| text_mode_bits | string | Specifies the textual format of mode bits on this file or directory.   |
| user_id        | string | Specifies user ID of this file or directory.   |

href

| Name | Type   | Description |
|------|--------|-------------|
| href | string |             |

\_links

| Name | Type                 | Description |
|------|----------------------|-------------|
| self | <a href="#">href</a> |             |

job\_link

| Name   | Type                   | Description   |
|--------|------------------------|---|
| _links | <a href="#">_links</a> |   |
| uuid   | string                 | The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation. |

error\_arguments

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

returned\_error

| Name      | Type                                     | Description                                 |
|-----------|--|---|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments                           |
| code      | string                                   | Error code                                  |
| message   | string                                   | Error message                               |
| target    | string                                   | The target parameter that caused the error. |

## Add a new SACL or DACL ACE

POST /protocols/file-security/permissions/{svm.uuid}/{path}/acl

**Introduced In:** 9.9

Adds the new SACL/DACL ACE. You must keep the following points in mind while using these endpoints:

- SLAG applies to all files and/or directories in a volume hence, inheritance is not required to be propagated.
- Set `access_control` field to `slag` while adding SLAG ACE.
- Set `access_control` field to `file_directory` while adding file-directory ACE. By Default `access_control` field is set to `file_directory`.
- For SLAG, valid `apply_to` combinations are "this-folder, sub-folders", "files", "this-folder, sub-folders, files".

### Required properties

- `user` - Name of the user to which the ACE applies.
- `access` - Specifies whether the ACE is for DACL or SACL.

### Related ONTAP commands

- `vserver security file-directory ntfs dacl add`
- `vserver security file-directory ntfs sacl add`

### Parameters

| Name | Type   | In   | Required | Description |
|------|--------|------|----------|-------------|
| path | string | path | True     | path        |

| Name           | Type    | In    | Required | Description  |
|----------------|---------|-------|----------|--|
| return_timeout | integer | query | False    | <p>The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.</p> <ul style="list-style-type: none"> <li>• Default value: 1</li> <li>• Max value: 120</li> <li>• Min value: 0</li> </ul> |
| return_records | boolean | query | False    | <p>The default is false. If set to true, the records are returned.</p> <ul style="list-style-type: none"> <li>• Default value:</li> </ul>  |
| svm.uuid       | string  | path  | True     | <p>UUID of the SVM to which this object belongs.</p>   |

## Request Body

| Name            | Type                            | Description   |
|-----------------|---------------------------------|---|
| access          | string                          | <p>Specifies whether the ACL is for DACL or SACL. It is a required field. The available values are:</p> <ul style="list-style-type: none"> <li>• access_allow - DACL for allow access</li> <li>• access_deny - DACL for deny access</li> <li>• audit_success - SACL for success access</li> <li>• audit_failure - SACL for failure access</li> </ul>  |
| access_control  | string                          | <p>Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable).</p> |
| advanced_rights | <a href="#">advanced_rights</a> | <p>Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.</p>  |
| apply_to        | <a href="#">apply_to</a>        | <p>Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.</p>  |
| ignore_paths    | array[string]                   | <p>Specifies that permissions on this file or directory cannot be replaced.</p>   |

| Name             | Type   | Description  |
|------------------|--------|--|
| propagation_mode | string | <p>Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are:</p> <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• ignore - ignore inheritable permissions</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |
| rights           | string | <p>Specifies the access rights controlled by the ACE for the account specified. The "rights" parameter is mutually exclusive with the "advanced_rights" parameter. If you specify the "rights" parameter, you can specify one of the following "rights" values ("- " or " _ " is accepted as the delimiter).</p>   |
| user             | string | <p>Specifies the account to which the ACE applies. You can specify either name or SID.</p>   |

## Example request

```
{
  "access": "access_allow",
  "access_control": "file_directory",
  "ignore_paths": [
    "/dir1/dir2/",
    "/parent/dir3"
  ],
  "propagation_mode": "string",
  "rights": "full_control",
  "user": "S-1-5-21-2233347455-2266964949-1780268902-69304"
}
```

## Response

Status: 202, Accepted

| Name | Type     | Description |
|------|----------|-------------|
| job  | job_link |             |

## Example response

```
{
  "job": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "string"
  }
}
```

## Headers

| Name     | Description                               | Type   |
|----------|---|--------|
| Location | Useful for tracking the resource location | string |

## Response

Status: 201, Created

## Error

Status: Default

### ONTAP Error Response Codes

| Error Code | Description   |
|------------|---|
| 655865     | The specified file or directory does not exist.     |
| 10485811   | Access is a required field.                         |
| 1260882    | Specified SVM not found.                            |
| 6691623    | User is not authorized.                             |
| 4849676    | The specified Windows user or group does not exist. |
| 10485810   | User is a required field.                           |
| 10485813   | All values cannot be false.                         |

| Name  | Type                           | Description |
|-------|--------------------------------|-------------|
| error | <a href="#">returned_error</a> |             |

### Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

# Definitions



## See Definitions

### advanced\_rights

Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.

| Name         | Type    | Description               |
|--------------|---------|---------------------------|
| append_data  | boolean | Append DAta               |
| delete       | boolean | Delete                    |
| delete_child | boolean | Delete Child              |
| execute_file | boolean | Execute File              |
| full_control | boolean | Full Control              |
| read_attr    | boolean | Read Attributes           |
| read_data    | boolean | Read Data                 |
| read_ea      | boolean | Read Extended Attributes  |
| read_perm    | boolean | Read Permissions          |
| synchronize  | boolean | Synchronize               |
| write_attr   | boolean | Write Attributes          |
| write_data   | boolean | Write Data                |
| write_ea     | boolean | Write Extended Attributes |
| write_owner  | boolean | Write Owner               |
| write_perm   | boolean | Write Permission          |

### apply\_to

Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.

| Name  | Type    | Description    |
|-------|---------|----------------|
| files | boolean | Apply to Files |

| Name        | Type    | Description               |
|-------------|---------|---------------------------|
| sub_folders | boolean | Apply to all sub-folders  |
| this_folder | boolean | Apply only to this folder |

file\_directory\_security\_acl

Manages the DACLS or SACLs.

| Name            | Type                            | Description   |
|-----------------|---------------------------------|---|
| access          | string                          | <p>Specifies whether the ACL is for DACL or SACL. It is a required field. The available values are:</p> <ul style="list-style-type: none"> <li>• access_allow - DACL for allow access</li> <li>• access_deny - DACL for deny access</li> <li>• audit_success - SACL for success access</li> <li>• audit_failure - SACL for failure access</li> </ul>  |
| access_control  | string                          | <p>Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable).</p> |
| advanced_rights | <a href="#">advanced_rights</a> | <p>Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.</p>  |

| Name             | Type                     | Description   |
|------------------|--------------------------|---|
| apply_to         | <a href="#">apply_to</a> | Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.   |
| ignore_paths     | array[string]            | Specifies that permissions on this file or directory cannot be replaced.  |
| propagation_mode | string                   | Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are: <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• ignore - ignore inheritable permissions</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |
| rights           | string                   | Specifies the access rights controlled by the ACE for the account specified. The "rights" parameter is mutually exclusive with the "advanced_rights" parameter. If you specify the "rights" parameter, you can specify one of the following "rights" values ("- " or "_ " is accepted as the delimiter).  |
| user             | string                   | Specifies the account to which the ACE applies. You can specify either name or SID.   |

href

| Name | Type   | Description |
|------|--------|-------------|
| href | string |             |

\_links

| Name | Type                 | Description |
|------|----------------------|-------------|
| self | <a href="#">href</a> |             |

job\_link

| Name   | Type                   | Description   |
|--------|------------------------|---|
| _links | <a href="#">_links</a> |   |
| uuid   | string                 | The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation. |

error\_arguments

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

returned\_error

| Name      | Type                                     | Description                                 |
|-----------|--|---|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments                           |
| code      | string                                   | Error code                                  |
| message   | string                                   | Error message                               |
| target    | string                                   | The target parameter that caused the error. |

## Delete a SACL or DACL ACL

```
DELETE /protocols/file-security/permissions/{svm.uuid}/{path}/acl/{user}
```

**Introduced In:** 9.9

Deletes the SACL/DACL ACL You must keep the following points in mind while using these endpoints:

- SLAG applies to all files and/or directories in a volume hence, inheritance is not required to be propagated.
- Set `access_control` field to `slag` while deleting SLAG ACE.
- Set `access_control` field to `file_directory` while deleting file-directory ACE. By Default `access_control` field is

set to `file_directory`.

- For SLAG, valid `apply_to` combinations are "this-folder, sub-folders", "files", "this-folder, sub-folders, files".

## Required properties

- `access` - Specifies whether the ACE is for DACL or SACL.
- `user` - Name of the user to which the ACE applies.

## Related ONTAP commands

- `vserver security file-directory ntfs dacl remove`
- `vserver security file-directory ntfs sacl remove`

## Parameters

| Name                        | Type    | In    | Required | Description   |
|-----------------------------|---------|-------|----------|---|
| <code>path</code>           | string  | path  | True     | path  |
| <code>user</code>           | string  | path  | True     | User Name   |
| <code>return_records</code> | boolean | query | False    | The default is false. If set to true, the records are returned.<br><br>• Default value: |

| Name           | Type    | In    | Required | Description  |
|----------------|---------|-------|----------|--|
| return_timeout | integer | query | False    | <p>The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.</p> <ul style="list-style-type: none"> <li>• Default value: 1</li> <li>• Max value: 120</li> <li>• Min value: 0</li> </ul> |
| svm.uuid       | string  | path  | True     | <p>UUID of the SVM to which this object belongs.</p>   |

## Request Body

| Name           | Type                     | Description  |
|----------------|--------------------------|--|
| access         | string                   | <p>Specifies whether the ACL is for DACL or SACL. It is a required field. The available values are:</p> <ul style="list-style-type: none"> <li>• access_allow - DACL for allow access</li> <li>• access_deny - DACL for deny access</li> <li>• audit_success - SACL for success access</li> <li>• audit_failure - SACL for failure access</li> </ul>   |
| access_control | string                   | <p>An Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable).</p> |
| apply_to       | <a href="#">apply_to</a> | <p>Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.</p>   |
| ignore_paths   | array[string]            | <p>Specifies that permissions on this file or directory cannot be replaced.</p>  |

| Name             | Type   | Description   |
|------------------|--------|---|
| propagation_mode | string | <p>Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are:</p> <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |

### Example request

```
{
  "access": "access_allow",
  "access_control": "file_directory",
  "ignore_paths": [
    "/dir1/dir2/",
    "/parent/dir3"
  ],
  "propagation_mode": "string"
}
```

### Response

Status: 200, Ok

| Name | Type     | Description |
|------|----------|-------------|
| job  | job_link |             |



## Example response

```
{
  "job": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "string"
  }
}
```

## Response

Status: 202, Accepted

## Error

Status: Default

### ONTAP Error Response Codes

| Error Code | Description   |
|------------|---|
| 655865     | The specified file or directory does not exist.     |
| 10485811   | Access is a required field.                         |
| 1260882    | Specified SVM not found.                            |
| 6691623    | User is not authorized.                             |
| 4849676    | The specified Windows user or group does not exist. |
| 10485813   | All values cannot be false.                         |

| Name  | Type                           | Description |
|-------|--------------------------------|-------------|
| error | <a href="#">returned_error</a> |             |

## Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

## Definitions

## See Definitions

### apply\_to

Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.

| Name        | Type    | Description               |
|-------------|---------|---------------------------|
| files       | boolean | Apply to Files            |
| sub_folders | boolean | Apply to all sub-folders  |
| this_folder | boolean | Apply only to this folder |

### acl\_delete

Manages the DACLS or SACLs.

| Name           | Type   | Description   |
|----------------|--------|---|
| access         | string | Specifies whether the ACL is for DACL or SACL. It is a required field. The available values are: <ul style="list-style-type: none"><li>• access_allow - DACL for allow access</li><li>• access_deny - DACL for deny access</li><li>• audit_success - SACL for success access</li><li>• audit_failure - SACL for failure access</li></ul>  |
| access_control | string | An Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable). |

| Name             | Type                     | Description  |
|------------------|--------------------------|--|
| apply_to         | <a href="#">apply_to</a> | Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.  |
| ignore_paths     | array[string]            | Specifies that permissions on this file or directory cannot be replaced.   |
| propagation_mode | string                   | Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are: <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |

href

| Name | Type   | Description |
|------|--------|-------------|
| href | string |             |

\_links

| Name | Type                 | Description |
|------|----------------------|-------------|
| self | <a href="#">href</a> |             |

job\_link

| Name   | Type                   | Description   |
|--------|------------------------|---|
| _links | <a href="#">_links</a> |   |
| uuid   | string                 | The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation. |

error\_arguments

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

returned\_error

| Name      | Type                                     | Description                                 |
|-----------|--|---|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments                           |
| code      | string                                   | Error code                                  |
| message   | string                                   | Error message                               |
| target    | string                                   | The target parameter that caused the error. |

## Update SACLS or DACLS

PATCH /protocols/file-security/permissions/{svm.uuid}/{path}/acl/{user}

**Introduced In:** 9.9

Updates the SACLS/DACLs You must keep the following points in mind while using these endpoints:

- SLAG applies to all files and/or directories in a volume hence, inheritance is not required to be propagated.
- Set `access_control` field to `slag` while updating SLAG ACE.
- Set `access_control` field to `file_directory` while updating file-directory ACE. By Default `access_control` field is set to `file_directory`.
- For SLAG, valid `apply_to` combinations are "this-folder, sub-folders", "files", "this-folder, sub-folders, files".

### Required properties

- `access` - Specifies whether the ACE is for DACL or SACL.
- `user` - Name of the user to which the ACE applies.

### Related ONTAP commands

- `vserver security file-directory ntfs dacl modify`
- `vserver security file-directory ntfs sacl modify`

## Parameters

| Name           | Type    | In    | Required | Description  |
|----------------|---------|-------|----------|--|
| path           | string  | path  | True     | path   |
| user           | string  | path  | True     | User Name  |
| return_records | boolean | query | False    | <p>The default is false. If set to true, the records are returned.</p> <ul style="list-style-type: none"> <li>• Default value:</li> </ul>  |
| return_timeout | integer | query | False    | <p>The number of seconds to allow the call to execute before returning. When doing a POST, PATCH, or DELETE operation on a single record, the default is 0 seconds. This means that if an asynchronous operation is started, the server immediately returns HTTP code 202 (Accepted) along with a link to the job. If a non-zero value is specified for POST, PATCH, or DELETE operations, ONTAP waits that length of time to see if the job completes so it can return something other than 202.</p> <ul style="list-style-type: none"> <li>• Default value: 1</li> <li>• Max value: 120</li> <li>• Min value: 0</li> </ul> |
| svm.uuid       | string  | path  | True     | UUID of the SVM to which this object belongs.  |

## Request Body

| Name            | Type                            | Description   |
|-----------------|---------------------------------|---|
| access          | string                          | <p>Specifies whether the ACL is for DACL or SACL. It is a required field. The available values are:</p> <ul style="list-style-type: none"> <li>• access_allow - DACL for allow access</li> <li>• access_deny - DACL for deny access</li> <li>• audit_success - SACL for success access</li> <li>• audit_failure - SACL for failure access</li> </ul>  |
| access_control  | string                          | <p>Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable).</p> |
| advanced_rights | <a href="#">advanced_rights</a> | <p>Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.</p>  |
| apply_to        | <a href="#">apply_to</a>        | <p>Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.</p>  |
| ignore_paths    | array[string]                   | <p>Specifies that permissions on this file or directory cannot be replaced.</p>   |

| Name             | Type   | Description  |
|------------------|--------|--|
| propagation_mode | string | <p>Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are:</p> <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• ignore - ignore inheritable permissions</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |
| rights           | string | <p>Specifies the access rights controlled by the ACE for the account specified. The "rights" parameter is mutually exclusive with the "advanced_rights" parameter. If you specify the "rights" parameter, you can specify one of the following "rights" values ("-" or "_" is accepted as the delimiter).</p>  |
| user             | string | <p>Specifies the account to which the ACE applies. You can specify either name or SID.</p>   |



## Example request

```
{
  "access": "access_allow",
  "access_control": "file_directory",
  "ignore_paths": [
    "/dir1/dir2/",
    "/parent/dir3"
  ],
  "propagation_mode": "string",
  "rights": "full_control",
  "user": "S-1-5-21-2233347455-2266964949-1780268902-69304"
}
```

## Response

Status: 200, Ok

| Name | Type     | Description |
|------|----------|-------------|
| job  | job_link |             |

## Example response

```
{
  "job": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "uuid": "string"
  }
}
```

## Response

Status: 202, Accepted

## Error

Status: Default

### ONTAP Error Response Codes

| Error Code | Description   |
|------------|---|
| 655865     | The specified file or directory does not exist.     |
| 10485811   | Access is a required field.                         |
| 1260882    | Specified SVM not found.                            |
| 6691623    | User is not authorized.                             |
| 4849676    | The specified Windows user or group does not exist. |
| 10485813   | All values cannot be false.                         |

| Name  | Type                           | Description |
|-------|--------------------------------|-------------|
| error | <a href="#">returned_error</a> |             |

### Example error

```
{
  "error": {
    "arguments": [
      {
        "code": "string",
        "message": "string"
      }
    ],
    "code": "4",
    "message": "entry doesn't exist",
    "target": "uuid"
  }
}
```

## Definitions

## See Definitions

### advanced\_rights

Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.

| Name         | Type    | Description               |
|--------------|---------|---------------------------|
| append_data  | boolean | Append DAta               |
| delete       | boolean | Delete                    |
| delete_child | boolean | Delete Child              |
| execute_file | boolean | Execute File              |
| full_control | boolean | Full Control              |
| read_attr    | boolean | Read Attributes           |
| read_data    | boolean | Read Data                 |
| read_ea      | boolean | Read Extended Attributes  |
| read_perm    | boolean | Read Permissions          |
| synchronize  | boolean | Synchronize               |
| write_attr   | boolean | Write Attributes          |
| write_data   | boolean | Write Data                |
| write_ea     | boolean | Write Extended Attributes |
| write_owner  | boolean | Write Owner               |
| write_perm   | boolean | Write Permission          |

### apply\_to

Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.

| Name  | Type    | Description    |
|-------|---------|----------------|
| files | boolean | Apply to Files |

| Name        | Type    | Description               |
|-------------|---------|---------------------------|
| sub_folders | boolean | Apply to all sub-folders  |
| this_folder | boolean | Apply only to this folder |

file\_directory\_security\_acl

Manages the DACLS or SACLs.

| Name            | Type                            | Description  |
|-----------------|---------------------------------|--|
| access          | string                          | Specifies whether the ACL is for DACL or SACL. It is a required field. The available values are: <ul style="list-style-type: none"> <li>• access_allow - DACL for allow access</li> <li>• access_deny - DACL for deny access</li> <li>• audit_success - SACL for success access</li> <li>• audit_failure - SACL for failure access</li> </ul>  |
| access_control  | string                          | Access Control Level specifies the access control of the task to be applied. Valid values are "file-directory" or "Storage-Level Access Guard (SLAG)". SLAG is used to apply the specified security descriptors with the task for the volume or qtree. Otherwise, the security descriptors are applied on files and directories at the specified path. The value SLAG is not supported on FlexGroups volumes. The default value is "file-directory" ('-' and '_' are interchangeable). |
| advanced_rights | <a href="#">advanced_rights</a> | Specifies the advanced access right controlled by the ACE for the account specified. You can specify more than one "advanced-rights" value by using a comma-delimited list.  |

| Name             | Type                     | Description   |
|------------------|--------------------------|---|
| apply_to         | <a href="#">apply_to</a> | Specifies where to apply the DACL or SACL entries. You can specify more than one value by using a comma-delimited list.   |
| ignore_paths     | array[string]            | Specifies that permissions on this file or directory cannot be replaced.  |
| propagation_mode | string                   | Specifies how to propagate security settings to child subfolders and files. This setting determines how child files/folders contained within a parent folder inherit access control and audit information from the parent folder. The available values are: <ul style="list-style-type: none"> <li>• propagate - propagate inheritable permissions to all subfolders and files</li> <li>• ignore - ignore inheritable permissions</li> <li>• replace - replace existing permissions on all subfolders and files with inheritable permissions</li> </ul> |
| rights           | string                   | Specifies the access rights controlled by the ACE for the account specified. The "rights" parameter is mutually exclusive with the "advanced_rights" parameter. If you specify the "rights" parameter, you can specify one of the following "rights" values ("- " or "_ " is accepted as the delimiter).  |
| user             | string                   | Specifies the account to which the ACE applies. You can specify either name or SID.   |

href

| Name | Type   | Description |
|------|--------|-------------|
| href | string |             |

## `_links`

| Name | Type                 | Description |
|------|----------------------|-------------|
| self | <a href="#">href</a> |             |

## `job_link`

| Name                | Type                   | Description   |
|---------------------|------------------------|---|
| <code>_links</code> | <a href="#">_links</a> |   |
| uuid                | string                 | The UUID of the asynchronous job that is triggered by a POST, PATCH, or DELETE operation. |

## `error_arguments`

| Name    | Type   | Description      |
|---------|--------|------------------|
| code    | string | Argument code    |
| message | string | Message argument |

## `returned_error`

| Name      | Type                                     | Description                                 |
|-----------|--|---|
| arguments | array[ <a href="#">error_arguments</a> ] | Message arguments                           |
| code      | string                                   | Error code                                  |
| message   | string                                   | Error message                               |
| target    | string                                   | The target parameter that caused the error. |

## Copyright information

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

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

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

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

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

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

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

## Trademark information

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