# Manage CIFS UNIX symlink mapping

ONTAP 9.12.1 REST API reference

NetApp
February 13, 2024

# Table of Contents

# Manage CIFS UNIX symlink mapping

## Protocols CIFS unix-symlink-mapping endpoint overview

### Overview

ONTAP allows both CIFS and NFS to access the same datastore. This datastore can contain symbolic links which are files, created by UNIX clients. It contains a reference to another file or directory. If an SMB client accesses a symbolic link, it is redirected to the target file or directory that the symbolic link refers to. The symbolic links can point to files within the volume that contain the share, or to files that are contained in other volumes on the Storage Virtual Machine (SVM), or even to volumes contained on other SVMs.

There are two types of symbolic links:

- 

**Relative** A relative symbolic link contains a reference to the file or directory relative to its parent directory. Therefore, the path of the file it is referring to should not begin with a backslash (/). If you enable symbolic links on a share, relative symbolic links work without UNIX symlink mapping.

- 

**Absolute** An absolute symbolic link contains a reference to a file or directory in the form of an absolute path. Therefore, the path of the file it is referring to should begin with a backslash (/). An absolute symbolic link can refer to a file or directory within or outside of the file system of the symbolic link. If the target is not in the same local file system, the symbolic link is called a "widelink". If the symbolic link is enabled on a share and absolute symbolic links do not work right away, the mapping between the UNIX path of the symbolic link to the destination CIFS path must be created. When creating absolute symbolic link mappings, locality could be either "local" or "widelink" and it must be specified. If UNIX symlink mapping is created for a file or directory which is outside of the local share but the locality is set to "local", ONTAP does not allow access to the target.

A UNIX symbolic link support could be added to SMB shares by specifying the *unix_symlink* property during the creation of SMB shares or at any time by modifying the existing SMB *unix_symlink* property. UNIX symbolic link support is enabled by default.

### Examples

#### Creating a UNIX symlink mapping for CIFS shares

To create UNIX symlink mappings for SMB shares, use the following API. Note the *return_records=true* query parameter used to obtain the newly created entry in the response.

```
# The API:
POST /api/protocols/cifs/unix-symlink-mapping

# The call:
curl -X POST "https://<mgmt-ip>/api/protocols/cifs/unix-symlink-
mapping?return_records=true" -H "accept: application/json" -H "Content-
Type: application/json" -d "{ \"svm\": { \"name\": \"vs1\", \"uuid\":
\"000c5cd2-ebdf-11e8-a96e-0050568ea3cb\" }, \"target\": {
\"home_directory\": false, \"locality\": \"local\", \"path\":
\"/dir1/dir2/\", \"server\": \"cifs123\", \"share\": \"sh1\" },
\"unix_path\": \"/mnt/eng_volume/\"}"

# The response:
{
"num_records": 1,
"records": [
  {
    "svm": {
      "uuid": "000c5cd2-ebdf-11e8-a96e-0050568ea3cb",
      "name": "vs1"
    },
    "unix_path": "/mnt/eng_volume/",
    "target": {
      "share": "sh1",
      "path": "/dir1/dir2/",
      "server": "cifs123",
      "locality": "local",
      "home_directory": false
    }
  }
]
}
```

**Retrieving UNIX symlink mappings for all SVMs in the cluster**

```
# The API:
GET /api/protocols/cifs/unix-symlink-mapping

# The call:
curl -X GET "https://<mgmt-ip>/api/protocols/cifs/unix-symlink-
mapping?fields=*&return_records=true&return_timeout=15" -H "accept:
application/hal+json"
```

```
# The response:
{
"records": [
  {
    "svm": {
      "uuid": "000c5cd2-ebdf-11e8-a96e-0050568ea3cb",
      "name": "vs1",
      "_links": {
        "self": {
          "href": "/api/svm/svms/000c5cd2-ebdf-11e8-a96e-0050568ea3cb"
        }
      }
    },
    "unix_path": "/mnt/eng_volume/",
    "target": {
      "share": "sh1",
      "path": "/dir1/dir2/",
      "server": "CIFS123",
      "locality": "local",
      "home_directory": false
    },
    "_links": {
      "self": {
        "href": "/api/protocols/cifs/unix-symlink-mapping/000c5cd2-ebdf-
11e8-a96e-0050568ea3cb/%2Fmnt%2Feng_volume%2F"
      }
    }
  },
  {
    "svm": {
      "uuid": "1d30d1b1-ebdf-11e8-a96e-0050568ea3cb",
      "name": "vs2",
      "_links": {
        "self": {
          "href": "/api/svm/svms/1d30d1b1-ebdf-11e8-a96e-0050568ea3cb"
        }
      }
    },
    "unix_path": "/mnt/eng_volume/",
    "target": {
      "share": "ENG_SHARE",
      "path": "/dir1/dir2/",
      "server": "ENGCIFS",
      "locality": "widelink",
      "home_directory": false
    },
```

```
    "_links": {
      "self": {
        "href": "/api/protocols/cifs/unix-symlink-mapping/1d30d1b1-ebdf-
11e8-a96e-0050568ea3cb/%2Fmnt%2Feng_volume%2F"
      }
    }
  }
],
"num_records": 2,
"_links": {
  "self": {
    "href": "/api/protocols/cifs/unix-symlink-
mapping?fields=*&return_records=true&return_timeout=15"
  }
}
}
```

**Retrieving a specific UNIX symlink mapping for an SVM**

The mapping being returned is identified by the UUID of its SVM and the unix-path.

```
# The API:
GET /api/protocols/cifs/unix-symlink-mapping/{svm.uuid}/{unix_path}

# The call:
curl -X GET "https://<mgmt-ip>/api/protocols/cifs/unix-symlink-
mapping/000c5cd2-ebdf-11e8-a96e-0050568ea3cb/%2Fmnt%2Feng_volume%2F" -H
"accept: application/json"

# The response:
{
"svm": {
  "uuid": "000c5cd2-ebdf-11e8-a96e-0050568ea3cb",
 "name": "vs1"
},
"unix_path": "/mnt/eng_volume/",
"target": {
  "share": "sh1",
  "path": "/dir1/dir2/",
  "server": "CIFS123",
  "locality": "local",
  "home_directory": false
}
}
```

#### Updating a specific UNIX symlink mapping for an SVM

The mapping being modified is identified by the UUID of its SVM and the unix-path.

```
# The API:
PATCH /api/protocols/cifs/unix-symlink-mapping/{svm.uuid}/{unix_path}

# The call:
curl -X PATCH "https://<mgmt-ip>/api/protocols/cifs/unix-symlink-
mapping/000c5cd2-ebdf-11e8-a96e-0050568ea3cb/%2Fmnt%2Feng_volume%2F" -H
"accept: application/json" -H "Content-Type: application/json" -d "{
\"target\": { \"home_directory\": true, \"locality\": \"widelink\",
\"path\": \"/new_path/\", \"server\": \"HR_SERVER\", \"share\": \"sh2\"
}}"
```

#### Removing a specific UNIX symlink mapping for an SVM

The mapping being removed is identified by the UUID of its SVM and the unix-path.

```
# The API:
DELETE /api/protocols/cifs/unix-symlink-mapping/{svm.uuid}/{unix_path}

# The call:
curl -X DELETE "https://<mgmt-ip>/api/protocols/cifs/unix-symlink-
mapping/000c5cd2-ebdf-11e8-a96e-0050568ea3cb/%2Fmnt%2Feng_volume%2F" -H
"accept: application/json"
```

# Retrieve a UNIX symbolic link mappings for CIFS clients

GET `/protocols/cifs/unix-symlink-mapping`

**Introduced In:** 9.6

Retrieves UNIX symbolic link mappings for CIFS clients.

## Related ONTAP commands

- `vserver cifs symlink show`

## Learn more

- DOC /protocols/cifs/unix-symlink-mapping

## Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| target.server | string | query | False | Filter by target.server<br>• maxLength: 45 |
| target.locality | string | query | False | Filter by target.locality |
| target.share | string | query | False | Filter by target.share<br>• maxLength: 80 |
| target.home_directory | boolean | query | False | Filter by target.home_directory |
| target.path | string | query | False | Filter by target.path<br>• maxLength: 256 |
| svm.uuid | string | query | False | Filter by svm.uuid |
| svm.name | string | query | False | Filter by svm.name |
| unix_path | string | query | False | Filter by unix_path<br>• maxLength: 256 |
| fields | array[string] | query | False | Specify the fields to return. |
| max_records | integer | query | False | Limit the number of records returned. |
| return_records | boolean | query | False | The default is true for GET calls. When set to false, only the number of records is returned.<br>• Default value: 1 |

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| return_timeout | integer | query | False | The number of seconds to allow the call to execute before returning. When iterating over a collection, the default is 15 seconds. ONTAP returns earlier if either max records or the end of the collection is reached.<br>• Default value: 1<br>• Max value: 120<br>• Min value: 0 |
| order_by | array[string] | query | False | Order results by specified fields and optional [asc |

## Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| num_records | integer | Number of records |
| records | array[cifs_symlink_mapping] | |

**Example response**


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "svm": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "svm1",
      "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
    },
    "target": {
      "locality": "local",
      "path": "/dir1/dir2/",
      "server": "ENGCIFS",
      "share": "ENG_SHARE"
    },
    "unix_path": "/mnt/eng_volume/"
  }
}
```


## Error

```
Status: Default, Error
```

| Name | Type | Description |
| --- | --- | --- |
| error | error | |

**Example error**


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


## Definitions

**See Definitions**

href

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

_links

| Name | Type | Description |
|---|---|---|
| next | href | |
| self | href | |

_links

| Name | Type | Description |
|---|---|---|
| self | href | |

svm

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. |
| uuid | string | The unique identifier of the SVM. |

cifs_target

| Name | Type | Description |
|---|---|---|
| home_directory | boolean | Specify if the destination share is a home directory. |
| locality | string | Specifies whether the CIFS symbolic link is a local link or wide link. The following values are supported:<br>• local - Local symbolic link maps only to the same CIFS share.<br>• widelink - Wide symbolic link maps to any CIFS share on the network. |

| Name | Type | Description |
| --- | --- | --- |
| path | string | Specifies the CIFS path on the destination to which the symbolic link maps. The final path is generated by concatenating the CIFS server name, the share name, the cifs-path and the remaining path in the symbolic link left after the prefix match. This value is specified by using a UNIX-style path name. The trailing forward slash is required for the full path name to be properly interpreted. |
| server | string | Specifies the destination CIFS server where the UNIX symbolic link is pointing. This field is mandatory if the locality of the symbolic link is 'widelink'. You can specify the value in any of the following formats:<br>• DNS name of the CIFS server.<br>• IP address of the CIFS server.<br>• NetBIOS name of the CIFS server. |
| share | string | Specifies the CIFS share name on the destination CIFS server to which the UNIX symbolic link is pointing. |

cifs_symlink_mapping

ONTAP allows for both CIFS and NFS access to the same datastore. This datastore can contain symbolic links created by UNIX clients which can point anywhere from the perspective of the UNIX client. To Access such UNIX symlink from CIFS share, we need to create a CIFS symbolic link path mapping from a UNIX symlink and target it as a CIFS path.

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |
| svm | svm | |
| target | cifs_target | |

| Name | Type | Description |
|---|---|---|
| unix_path | string | Specifies the UNIX path prefix to be matched for the mapping. |

error_arguments

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

error

| Name | Type | Description |
|---|---|---|
| arguments | array[error_arguments] | Message arguments |
| code | string | Error code |
| message | string | Error message |
| target | string | The target parameter that caused the error. |

# Create a UNIX symbolic link mapping for a CIFS client

POST `/protocols/cifs/unix-symlink-mapping`

**Introduced In:** 9.6

Creates a UNIX symbolic link mapping for a CIFS client.

## Required properties

- `svm.uuid` or `svm.name` - Existing SVM in which to create the CIFS unix-symlink-mapping.
- `unix_path` - UNIX path to which the CIFS symlink mapping to be created.
- `target.share` - CIFS share name on the destination CIFS server to which the UNIX symbolic link is pointing.
- `target.path` - CIFS path on the destination to which the symbolic link maps.

## Default property values

- `target.server` - *Local_NetBIOS_Server_Name*

- `locality` - *local*
- `home_directory` - *false*

## Related ONTAP commands

- `vserver cifs symlink create`

## Learn more

- DOC /protocols/cifs/unix-symlink-mapping

## Parameters

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

## Request Body

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| svm | svm | |
| target | cifs_target | |
| unix_path | string | Specifies the UNIX path prefix to be matched for the mapping. |

**Example request**

```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target": {
    "locality": "local",
    "path": "/dir1/dir2/",
    "server": "ENGCIFS",
    "share": "ENG_SHARE"
  },
  "unix_path": "/mnt/eng_volume/"
}
```

## Response

```
Status: 201, Created
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| num_records | integer | Number of records |
| records | array[cifs_symlink_mapping] | |

### Example response


```
{
  "_links": {
    "next": {
      "href": "/api/resourcelink"
    },
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "num_records": 1,
  "records": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "svm": {
      "_links": {
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "svm1",
      "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
    },
    "target": {
      "locality": "local",
      "path": "/dir1/dir2/",
      "server": "ENGCIFS",
      "share": "ENG_SHARE"
    },
    "unix_path": "/mnt/eng_volume/"
  }
}
```


### Headers

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

## Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 655654 | Must specify the target CIFS share while creating path mapping entries with localities "local" or "widelink" |
| 655572 | The target path contains illegal characters or is too long |
| 655574 | The target server contains illegal characters or is too long |
| 655436 | If the locality is "local", the target server must be blank or must match the CIFS NetBIOS name for given SVM |
| 655439 | The Specified target server is local CIFS server for given SVM but the locality is specified as "widelink" |
| 655546 | Failed to create symlink mapping becasue administrative share cannot be used as target share |
| 655437 | Failed to create the symlink mapping with locality "local" because the target share does not exist for specified SVM |
| 655429 | UNIX path must begin and end with a "/" |
| 655430 | Target path must begin and end with a "/" |
| 655399 | Failed to get the CIFS server for specified SVM |

| Name | Type | Description |
|---|---|---|
| error | error | |

**Example error**


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


## Definitions

**See Definitions**

href

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

_links

| Name | Type | Description |
|---|---|---|
| self | href | |

svm

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. |
| uuid | string | The unique identifier of the SVM. |

cifs_target

| Name | Type | Description |
|---|---|---|
| home_directory | boolean | Specify if the destination share is a home directory. |
| locality | string | Specifies whether the CIFS symbolic link is a local link or wide link. The following values are supported:<br>• local - Local symbolic link maps only to the same CIFS share.<br>• widelink - Wide symbolic link maps to any CIFS share on the network. |

| Name | Type | Description |
| --- | --- | --- |
| path | string | Specifies the CIFS path on the destination to which the symbolic link maps. The final path is generated by concatenating the CIFS server name, the share name, the cifs-path and the remaining path in the symbolic link left after the prefix match. This value is specified by using a UNIX-style path name. The trailing forward slash is required for the full path name to be properly interpreted. |
| server | string | Specifies the destination CIFS server where the UNIX symbolic link is pointing. This field is mandatory if the locality of the symbolic link is 'widelink'. You can specify the value in any of the following formats:<br>• DNS name of the CIFS server.<br>• IP address of the CIFS server.<br>• NetBIOS name of the CIFS server. |
| share | string | Specifies the CIFS share name on the destination CIFS server to which the UNIX symbolic link is pointing. |

cifs_symlink_mapping

ONTAP allows for both CIFS and NFS access to the same datastore. This datastore can contain symbolic links created by UNIX clients which can point anywhere from the perspective of the UNIX client. To Access such UNIX symlink from CIFS share, we need to create a CIFS symbolic link path mapping from a UNIX symlink and target it as a CIFS path.

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |
| svm | svm | |
| target | cifs_target | |

| Name | Type | Description |
|---|---|---|
| unix_path | string | Specifies the UNIX path prefix to be matched for the mapping. |

_links

| Name | Type | Description |
|---|---|---|
| next | href | |
| self | href | |

error_arguments

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

error

| Name | Type | Description |
|---|---|---|
| arguments | array[error_arguments] | Message arguments |
| code | string | Error code |
| message | string | Error message |
| target | string | The target parameter that caused the error. |

# Delete a UNIX symbolic link mapping for CIFS clients

DELETE `/protocols/cifs/unix-symlink-mapping/{svm.uuid}/{unix_path}`

**Introduced In:** 9.6

Deletes the UNIX symbolic link mapping for CIFS clients.

## Related ONTAP commands

- `vserver cifs symlink delete`

## Learn more

- DOC /protocols/cifs/unix-symlink-mapping

## Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| unix_path | string | path | True | UNIX symbolic link path |
| svm.uuid | string | path | True | UUID of the SVM to which this object belongs. |

## Response

```
Status: 200, Ok
```

## Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | error | |

**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 |

error

| Name | Type | Description |
|---|---|---|
| arguments | array[error_arguments] | Message arguments |
| code | string | Error code |
| message | string | Error message |
| target | string | The target parameter that caused the error. |

# Retrieve a UNIX symbolic link mapping for CIFS clients

GET `/protocols/cifs/unix-symlink-mapping/{svm.uuid}/{unix_path}`

**Introduced In:** 9.6

Retrieves a UNIX symbolic link mapping for CIFS clients.

## Related ONTAP commands

- `vserver cifs symlink show`

## Learn more

- DOC /protocols/cifs/unix-symlink-mapping

## Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| unix_path | string | path | True | UNIX symbolic link path |

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

## Response

```
Status: 200, Ok
```

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| svm | svm | |
| target | cifs_target | |
| unix_path | string | Specifies the UNIX path prefix to be matched for the mapping. |

### Example response


```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target": {
    "locality": "local",
    "path": "/dir1/dir2/",
    "server": "ENGCIFS",
    "share": "ENG_SHARE"
  },
  "unix_path": "/mnt/eng_volume/"
}
```


## Error

```
Status: Default, Error
```

| Name | Type | Description |
|---|---|---|
| error | error | |

**Example error**


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


## Definitions

**See Definitions**

href

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

_links

| Name | Type | Description |
|---|---|---|
| self | href | |

svm

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. |
| uuid | string | The unique identifier of the SVM. |

cifs_target

| Name | Type | Description |
|---|---|---|
| home_directory | boolean | Specify if the destination share is a home directory. |
| locality | string | Specifies whether the CIFS symbolic link is a local link or wide link. The following values are supported:<br>• local - Local symbolic link maps only to the same CIFS share.<br>• widelink - Wide symbolic link maps to any CIFS share on the network. |

| Name | Type | Description |
|---|---|---|
| path | string | Specifies the CIFS path on the destination to which the symbolic link maps. The final path is generated by concatenating the CIFS server name, the share name, the cifs-path and the remaining path in the symbolic link left after the prefix match. This value is specified by using a UNIX-style path name. The trailing forward slash is required for the full path name to be properly interpreted. |
| server | string | Specifies the destination CIFS server where the UNIX symbolic link is pointing. This field is mandatory if the locality of the symbolic link is 'widelink'. You can specify the value in any of the following formats:<br>• DNS name of the CIFS server.<br>• IP address of the CIFS server.<br>• NetBIOS name of the CIFS server. |
| share | string | Specifies the CIFS share name on the destination CIFS server to which the UNIX symbolic link is pointing. |

error_arguments

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

error

| Name | Type | Description |
|---|---|---|
| arguments | array[error_arguments] | Message arguments |

| Name | Type | Description |
|---|---|---|
| code | string | Error code |
| message | string | Error message |
| target | string | The target parameter that caused the error. |

# Update the UNIX symbolic link mapping for CIFS clients

PATCH `/protocols/cifs/unix-symlink-mapping/{svm.uuid}/{unix_path}`

**Introduced In:** 9.6

Updates the UNIX symbolic link mapping for CIFS clients.

## Related ONTAP commands

- `vserver cifs symlink modify`

## Learn more

- DOC /protocols/cifs/unix-symlink-mapping

## Parameters

| Name | Type | In | Required | Description |
|---|---|---|---|---|
| unix_path | string | path | True | UNIX symbolic link path |
| svm.uuid | string | path | True | UUID of the SVM to which this object belongs. |

## Request Body

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| svm | svm | |
| target | cifs_target | |
| unix_path | string | Specifies the UNIX path prefix to be matched for the mapping. |

**Example request**

```
{
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "svm": {
    "_links": {
      "self": {
        "href": "/api/resourcelink"
      }
    },
    "name": "svm1",
    "uuid": "02c9e252-41be-11e9-81d5-00a0986138f7"
  },
  "target": {
    "locality": "local",
    "path": "/dir1/dir2/",
    "server": "ENGCIFS",
    "share": "ENG_SHARE"
  },
  "unix_path": "/mnt/eng_volume/"
}
```

## Response

```
Status: 200, Ok
```

## Error

```
Status: Default
```

ONTAP Error Response Codes

| Error Code | Description |
|---|---|
| 655573 | Failed to modify the symlink mapping to target path because it contains illegal characters or is too long |
| 655575 | Failed to modify the symlink mapping to target server because it contains illegal characters or is too long |

| Error Code | Description |
| --- | --- |
| 655547 | Failed to modify symlink mapping becasue administrative share cannot be used as target share |

| Name | Type | Description |
| --- | --- | --- |
| error | error | |

**Example error**

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

## Definitions

**See Definitions**

href

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

_links

| Name | Type | Description |
|---|---|---|
| self | href | |

svm

| Name | Type | Description |
|---|---|---|
| _links | _links | |
| name | string | The name of the SVM. |
| uuid | string | The unique identifier of the SVM. |

cifs_target

| Name | Type | Description |
|---|---|---|
| home_directory | boolean | Specify if the destination share is a home directory. |
| locality | string | Specifies whether the CIFS symbolic link is a local link or wide link. The following values are supported:<br>• local - Local symbolic link maps only to the same CIFS share.<br>• widelink - Wide symbolic link maps to any CIFS share on the network. |

| Name | Type | Description |
| --- | --- | --- |
| path | string | Specifies the CIFS path on the destination to which the symbolic link maps. The final path is generated by concatenating the CIFS server name, the share name, the cifs-path and the remaining path in the symbolic link left after the prefix match. This value is specified by using a UNIX-style path name. The trailing forward slash is required for the full path name to be properly interpreted. |
| server | string | Specifies the destination CIFS server where the UNIX symbolic link is pointing. This field is mandatory if the locality of the symbolic link is 'widelink'. You can specify the value in any of the following formats:<br>• DNS name of the CIFS server.<br>• IP address of the CIFS server.<br>• NetBIOS name of the CIFS server. |
| share | string | Specifies the CIFS share name on the destination CIFS server to which the UNIX symbolic link is pointing. |

cifs_symlink_mapping

ONTAP allows for both CIFS and NFS access to the same datastore. This datastore can contain symbolic links created by UNIX clients which can point anywhere from the perspective of the UNIX client. To Access such UNIX symlink from CIFS share, we need to create a CIFS symbolic link path mapping from a UNIX symlink and target it as a CIFS path.

| Name | Type | Description |
| --- | --- | --- |
| _links | _links | |
| svm | svm | |
| target | cifs_target | |

| Name | Type | Description |
| --- | --- | --- |
| unix_path | string | Specifies the UNIX path prefix to be matched for the mapping. |

error_arguments

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

error

| Name | Type | Description |
| --- | --- | --- |
| arguments | array[error_arguments] | Message arguments |
| code | string | Error code |
| message | string | Error message |
| target | string | The target parameter that caused the error. |

## Copyright information



## Trademark information