Manage ONTAP with REST API

ONTAP Automation

NetApp
June 23, 2022

Table of Contents

Manage ONTAP with REST API 1
Manage EMS event notifications with the ONTAP REST API 1
Manage DACL and SACL file permissions with the ONTAP REST API 5

Manage ONTAP with REST API

Manage EMS event notifications with the ONTAP REST API

You can use the ONTAP REST API to configure how important Event Management System (EMS) event notifications are sent to you, as well as retrieve all EMS messages or specific messages with certain attributes. You can use the REST API to help automate EMS-related tasks, saving time over other methods of working with EMS messages.

The following examples show you how you can use the ONTAP REST API to work with EMS messages. In each example, be sure to provide your values for information in brackets <> where indicated. You can also reference a sample Python script that demonstrates how to automate a number of EMS-related activities.

View specific event logs

Using the `GET /support/ems/events` API call, you can retrieve specific event messages, such as the latest message, messages that contain specific text, or messages with a specific severity.

Example request: Retrieve the latest message

```
curl -X GET -u admin:<PASSWORD> -k
'https://<IP_ADDRESS>/api/support/ems/events?fields=message.name&max_recor
ds=1'
```

Example response


```
"records": [
    {
      "node": {
        "name": "malha-vsim1",
        "uuid": "da4f9e62-9de3-11ec-976a-005056b369de",
        "_links": {
          "self": {
            "href": "/api/cluster/nodes/da4f9e62-9de3-11ec-976a-
005056b369de"
          }
        }
      },
      "index": 6260,
      "message": {
        "name": "netinet6.rtr.high.mtu"
      },
      "_links": {
        "self": {
          "href": "/api/support/ems/events/malha-vsim1/6260"
        }
      }
    }
  ],
  "num_records": 1,
  "_links": {
    "self": {
      "href": "/api/support/ems/events?fields=message.name&max_records=1"
    },
    "next": {
      "href": "/api/support/ems/events?start.keytime=2022-03-
22T10%3A28%3A29-04%3A00&start.node.name=malha-
vsim1&start.index=6260&fields=message.name&max_records=1"
    }
  }
}
```


Example request: Retrieve a message containing specific text and severity

```
curl -X GET -u admin:<PASSWORD> -k
'https://<IP_ADDRESS>/api/support/ems/events?log_message=*disk*&message.se
verity=alert'
```

Example response


```
"records": [
    {
      "node": {
        "name": "malha-vsim1",
        "uuid": "da4f9e62-9de3-11ec-976a-005056b369de",
        "_links": {
          "self": {
            "href": "/api/cluster/nodes/da4f9e62-9de3-11ec-976a-
005056b369de"
          }
        }
      },
      "index": 4602,
      "time": "2022-03-18T06:37:46-04:00",
      "message": {
        "severity": "alert",
        "name": "raid.autoPart.disabled"
      },
      "log_message": "raid.autoPart.disabled: Disk auto-partitioning is
disabled on this system: the system needs a minimum of 4 usable internal
hard disks.",
      "_links": {
        "self": {
          "href": "/api/support/ems/events/malha-vsim1/4602"
        }
      }
    }
  ],
  "num_records": 1,
  "_links": {
    "self": {
      "href":
"/api/support/ems/events?log_message=*disk*&message.severity=alert&max_rec
ords=1"
    },
    "next": {
      "href": "/api/support/ems/events?start.keytime=2022-03-
18T06%3A37%3A46-04%3A00&start.node.name=malha-
vsim1&start.index=4602&log_message=*disk*&message.severity=alert"
    }
  }
}
```

View the existing EMS configuration

Using the `GET /support/ems/` API call, you can retrieve information about the existing EMS notification configuration.

Example request

```
curl -X GET -u admin:<PASSWORD> -k 'https://<IP_ADDRESS>/api/support/ems/'
```

Example response

```
{
  "proxy_url": "https://proxyserver.mycompany.com",
  "proxy_user": "proxy_user",
  "mail_server": "mail@mycompany.com",
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "pubsub_enabled": "1",
  "mail_from": "administrator@mycompany.com"
}
```

Create a new EMS notification configuration

You can use several API calls together to create a new EMS configuration to customize how and when you receive messages.

Steps

1. Configure the system-wide email settings. For example:

   ```
   curl -X PATCH -u admin:<PASSWORD> -k
   'https://<IP_ADDRESS>/api/support/ems?mail_from=administrator@mycompany.
   com&mail_server=mail@mycompany.com'
   ```

2. Define a filter that uses specific rules to match certain messages. For example:

   ```
   curl -u admin:<PASSWORD> -X POST -d '{"name": "test-filter",
   "rules.type": ["include"], "rules.message_criteria.severities":
   ["emergency"]}' -k 'https://<IP_ADDRESS>/api/support/ems/filters/'
   ```

3. Create a destination for the messages. For example:

```
curl -u admin:<PASSWORD> -X POST -d '{"name": "test-destination",
"type": "email", "destination": "administrator@mycompany.com",
"filters.name": ["important-events"]}' -k
'https://<IP_ADDRESS>/api/support/ems/destinations/'
```

ONTAP REST API versus ONTAP CLI commands

The ONTAP REST API enables you to automate workflows with fewer commands than the ONTAP CLI for many tasks. For example, you can use a single POST API method to create a filter, instead of using multiple CLI commands. The following table shows the CLI commands that you would need to complete common EMS tasks versus the corresponding REST API calls:

ONTAP REST API	ONTAP CLI
`GET /support/ems`	`event config show`
`POST /support/ems/destinations`	1. `event notification destination create` 2. `event notification create`
`GET /support/ems/events`	`event log show`
`POST /support/ems/filters`	1. `event filter create -filter-name <filtername>` 2. `event filter rule add -filter-name <filtername>`

Related information

- ONTAP REST API EMS example Python script
- ONTAP REST APIs: Automate Notification of High-Severity Events

Manage DACL and SACL file permissions with the ONTAP REST API

ONTAP uses System Access Control Lists (SACLs) and Discretionary Access Control Lists (DACLs) to assign permissions to file objects. Beginning with ONTAP 9.9.1, the ONTAP REST API includes endpoints to assign SACL and DACL permissions for files and automate file-security permissions. (You can learn more about options for automating SACL and DACL permissions prior to ONTAP 9.9.1 here).

Beginning with ONTAP 9.9.1, you can use a single REST API call in the place of multiple CLI commands or ONTAPI calls. The following examples show how you can use the ONTAP REST API to work with file permissions. In each example, be sure to provide your values for information in brackets <> where indicated.

You can also reference a sample Python script that demonstrates how to automate a number of SACL- and DACL-related activities.

View effective permissions

Using the `GET /protocols/file-security/effective-permissions/` API call, you can retrieve the current permissions for a specific file or directory.

Example request:

```
curl -X GET -u admin:<PASSWORD> -k
'https://<IP_ADDRESS>/api/protocols/file-security/effective-
permissions/cf5f271a-1beb-11ea-8fad-
005056bb645e/administrator/windows/%2F?share.name=sh1&return_records=true'
```

Example response


```
{
  "svm": {
    "uuid": "cf5f271a-1beb-11ea-8fad-005056bb645e",
    "name": "vs1"
  },
  "user": "administrator",
  "type": "windows",
  "path": "/",
  "share": {
    "path": "/"
  },
  "file_permission": [
    "read",
    "write",
    "append",
    "read_ea",
    "write_ea",
    "execute",
    "delete_child",
    "read_attributes",
    "write_attributes",
    "delete",
    "read_control",
    "write_dac",
    "write_owner",
    "synchronize",
    "system_security"
  ],
  "share_permission": [
    "read",
    "read_ea",
    "execute",
    "read_attributes",
    "read_control",
    "synchronize"
  ]
}
```


View all auditing information

Using the `GET /protocols/file-security/permissions/` API call, you can retrieve all auditing information for a specific file or directory.

Example request:

```
curl -X GET -u admin:<PASSWORD> -k
'https://<IP_ADDRESS>/api/protocols/file-security/permissions/9479099d-
5b9f-11eb-9c4e-0050568e8682/%2Fparent'
```

Example 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"
      }
    ],
    "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"
}
```


Apply new permissions

Using the `POST /protocols/file-security/permissions/` API call, you can apply a new security descriptor to a file or directory.

Example request

```
curl -u admin:<PASSWORD> -X POST -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\"}' -k 'https://<IP_ADDRESS>/api/protocols/file-
security/permissions/9479099d-5b9f-11eb-9c4e-
0050568e8682/%2Fparent?return_timeout=0'
```

Example response

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

Update security descriptor information

Using the `PATCH /protocols/file-security/permissions/` API call, you can update specific security descriptor information for a file or directory, such as the primary owner, group, or control flags.

Example request

```
curl -u admin:<PASSWORD> -X PATCH -d '{ \"control_flags\": \"32788\",
\"group\": \"everyone\", \"owner\": \"user1\"}' -k
'https://<IP_ADDRESS>/api/protocols/file-security/permissions/9479099d-
5b9f-11eb-9c4e-0050568e8682/%2Fparent?return_timeout=0'
```

Example response

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

Delete an existing SACL/DACL Access Control Entry (ACE)

Using the `DELETE /protocols/file-security/permissions/` API call, you can delete an existing ACE from a file for directory. This example propagates the change to any child objects.

Example request

```
curl -u admin:<PASSWORD> -X DELETE -d '{ \"access\": \"access_allow\",
\"apply_to\": { \"files\": true, \"sub_folders\": true, \"this_folder\":
true }, \"ignore_paths\": [ \"/parent/child2\" ], \"propagation_mode\":
\"propagate\"}' -k 'https://<IP_ADDRESS>/api/protocols/file-
security/permissions/9479099d-5b9f-11eb-9c4e-
0050568e8682/%2Fparent/acl/himanshu?return_timeout=0'
```

Example response

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

ONTAP REST API versus ONTAP CLI commands

The ONTAP REST API enables you to automate workflows with fewer commands than the ONTAP CLI for many tasks. For example, you can use a single POST API method to modify a security descriptor for a file, instead of using multiple CLI commands. The following table shows the CLI commands that you would need to complete common filesystem permission tasks versus the corresponding REST API calls:

ONTAP REST API	ONTAP CLI
`GET /protocols/file-security/effective-permissions/`	`vserver security file-directory show-effective-permissions`
`POST /protocols/file-security/permissions/`	1. `vserver security file-directory ntfs create` 2. `vserver security file-directory ntfs dacl add` 3. `vserver security file-directory ntfs sacl add` 4. `vserver security file-directory policy create` 5. `vserver security file-directory policy task add` 6. `vserver security file-directory apply`
`PATCH /protocols/file-security/permissions/`	`vserver security file-directory ntfs modify`
`DELETE /protocols/file-security/permissions/`	1. `vserver security file-directory ntfs dacl remove` 2. `vserver security file-directory ntfs sacl remove`

Related information

- ONTAP REST API DACL/SACL permissions example Python script
- Simplified management of file-security permissions with ONTAP REST APIs
- Using the private CLI passthrough with the ONTAP REST API (for versions of ONTAP prior to version 9.9.1)

Copyright Information

Trademark Information