NAS

ONTAP Automation

NetApp
July 11, 2024

目次

NAS

ファイルセキュリティ権限

ファイルセキュリティと監査ポリシーの管理の準備

ONTAPクラスタ内のSVMから使用可能なファイルの権限と監査ポリシーを管理できます。

概要

ONTAP では、システムアクセス制御リスト（ SACL ）と随意アクセス制御リスト（ DACL ）を使用してファイルオブジェクトに権限を割り当てます。ONTAP 9.9.1以降では、REST APIでSACLとDACLの権限の管理がサポートされます。APIを使用すると、ファイルセキュリティ権限の管理を自動化できます。多くの場合、複数のCLIコマンドやONTAPI（ZAPI）呼び出しではなく、1つのREST API呼び出しを使用できます。

> 9.9.1より前のONTAPリリースでは、CLIパススルー機能を使用して、SACLおよびDACL権限の管理を自動化できます。を参照してください "移行に関する考慮事項" および "ONTAP REST API でプライベート CLI パススルーを使用する" を参照してください。

REST APIを使用してONTAPファイルセキュリティサービスを管理する方法を示すワークフローの例をいくつか紹介します。ワークフローを使用してREST API呼び出しを実行する前に、 "ワークフローを使用する準備をします"。

Pythonを使用する場合は、スクリプトも参照してください。 "file_security_permissions.py" ファイルセキュリティアクティビティの一部を自動化する方法の例を参照してください。

ONTAP REST API コマンドと **ONTAP CLI** コマンドの比較

多くのタスクで、ONTAP REST APIを使用する場合、同等のONTAP CLIコマンドやONTAPI（ZAPI）呼び出しよりも少ない呼び出しで済みます。次の表に、API呼び出しと、各タスクに必要なCLIコマンドを示します。

ONTAP REST API	ONTAP CLI
「 get/protocols/file-security/effected-permissions/ 」	vserver security file-directory show-effected-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

ONTAP REST API	ONTAP CLI
'delete/protocols/file-security/permissions/	1. 「 vserver security file-directory ntfs dacl remove 」 2. 「 vserver security file-directory ntfs sacl remove 」

関連情報

- "ファイル権限を示すPythonスクリプト"
- "ONTAP REST API を使用してファイルセキュリティ権限を簡単に管理できます"
- "ONTAP REST API でプライベート CLI パススルーを使用する"

ファイルに有効な権限を取得する

特定のファイルまたはフォルダに対して現在有効な権限を取得できます。

HTTPメソッドとエンドポイント

このREST API呼び出しでは、次のメソッドとエンドポイントを使用します。

HTTP メソッド	パス
取得	/api/protocols/file-security/effective-permissions/｛svm.uuid｝/｛path｝

処理のタイプ

同期

curlの例の追加入力パラメータ

この手順のcurlの例では、すべてのREST API呼び出しに共通のパラメータに加えて、次のパラメータも使用しています。

パラメータ	を入力します	必須	説明
$SVM_ID	パス	はい。	これは、ファイルが含まれているSVMのUUIDです。
$file_path	パス	はい。	ファイルまたはフォルダへのパスです。

カールの例

```
curl --request GET \
--location "https://$FQDN_IP/api/protocols/file-security/effective-
permissions/$SVM_ID/$FILE_PATH" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 出力例


```
{
  "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"
  ]
}
```


ファイルの監査情報を取得する

特定のファイルまたはフォルダの監査情報を取得できます。

HTTPメソッドとエンドポイント

このREST API呼び出しでは、次のメソッドとエンドポイントを使用します。

HTTP メソッド	パス
取得	/api/protocols/file-security/permissions/｛svm.uuid｝/｛path｝

処理のタイプ

同期

curlの例の追加入力パラメータ

この手順のcurlの例では、すべてのREST API呼び出しに共通のパラメータに加えて、次のパラメータも使用しています。

パラメータ	を入力します	必須	説明
$SVM_ID	パス	はい。	これは、ファイルが含まれているSVMのUUIDです。
$file_path	パス	はい。	ファイルまたはフォルダへのパスです。

カールの例

```
curl --request GET \
--location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 出力例

```
{
  "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"
}
```

ファイルに新しい権限を適用する

新しいセキュリティ記述子を特定のファイルまたはフォルダに適用できます。

手順1：新しい権限を適用する

HTTPメソッドとエンドポイント

このREST API呼び出しでは、次のメソッドとエンドポイントを使用します。

HTTP メソッド	パス
投稿（ Post ）	/api/protocols/file-security/permissions/｛svm.uuid｝/｛path｝

処理のタイプ

非同期

curlの例の追加入力パラメータ

この手順のcurlの例では、すべてのREST API呼び出しに共通のパラメータに加えて、次のパラメータも使用しています。

パラメータ	を入力します	必須	説明
$SVM_ID	パス	はい。	これは、ファイルが含まれているSVMのUUIDです。
$file_path	パス	はい。	ファイルまたはフォルダへのパスです。

カールの例

```
curl --request POST --location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH?return_timeout=0" --include
--header "Accept */*" --header "Authorization: Basic $BASIC_AUTH" --data
'{ \"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\"}'
```

JSON 出力例

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

手順**2**：ジョブのステータスを取得する

ワークフローを実行 "ジョブインスタンスの取得" をクリックし、 `state` 値は `success`。

セキュリティ記述子情報を更新します

特定のセキュリティ記述子を、プライマリ所有者、グループ、制御フラグなど、特定のファイルまたはフォルダに対して更新できます。

手順**1**：セキュリティ記述子を更新する

HTTPメソッドとエンドポイント

このREST API呼び出しでは、次のメソッドとエンドポイントを使用します。

HTTP メソッド	パス
パッチ	/api/protocols/file-security/permissions/｛svm.uuid｝/｛path｝

処理のタイプ

非同期

curlの例の追加入力パラメータ

この手順のcurlの例では、すべてのREST API呼び出しに共通のパラメータに加えて、次のパラメータも使用しています。

パラメータ	を入力します	必須	説明
$SVM_ID	パス	はい。	これは、ファイルが含まれているSVMのUUIDです。
$file_path	パス	はい。	ファイルまたはフォルダへのパスです。

カールの例

```
curl --request POST --location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH?return_timeout=0" --include
--header "Accept */*" --header "Authorization: Basic $BASIC_AUTH" --data
'{ \"control_flags\": \"32788\", \"group\": \"everyone\", \"owner\":
\"user1\"}'
```

JSON 出力例

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

手順2：ジョブのステータスを取得する

ワークフローを実行 "ジョブインスタンスの取得" をクリックし、 `state` 値は `success`。

アクセス制御エントリを削除します。

特定のファイルまたはフォルダから既存のAccess Control Entry（ACE；アクセス制御エントリ）を削除できます。変更はすべての子オブジェクトに伝播されます。

手順1：ACEの削除

HTTPメソッドとエンドポイント

このREST API呼び出しでは、次のメソッドとエンドポイントを使用します。

HTTP メソッド	パス
削除	/api/protocols/file-security/permissions/｛svm.uuid｝/｛path｝

処理のタイプ

非同期

curlの例の追加入力パラメータ

この手順のcurlの例では、すべてのREST API呼び出しに共通のパラメータに加えて、次のパラメータも使用しています。

パラメータ	を入力します	必須	説明
$SVM_ID	パス	はい。	これは、ファイルが含まれているSVMのUUIDです。
$file_path	パス	はい。	ファイルまたはフォルダへのパスです。

カールの例

```
curl --request DELETE --location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH?return_timeout=0" --include
--header "Accept */*" --header "Authorization: Basic $BASIC_AUTH" --data
'{ \"access\": \"access_allow\", \"apply_to\": { \"files\": true,
\"sub_folders\": true, \"this_folder\": true }, \"ignore_paths\": [
\"/parent/child2\" ], \"propagation_mode\": \"propagate\"}'
```

JSON 出力例

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

手順2：ジョブのステータスを取得する

ワークフローを実行 "ジョブインスタンスの取得" をクリックし、 `state` 値は `success`。

著作権に関する情報

商標に関する情報