RBAC

ONTAP Automation

NetApp
July 25, 2024

目次

RBAC 1
RBACを使用するための準備 1
ロールの作成 1
ロールを持つユーザを作成します 5

RBAC

RBACを使用するための準備

ONTAP RBAC機能は、環境に応じていくつかの方法で使用できます。このセクションでは、いくつかの一般的なシナリオをワークフローとして紹介します。いずれの場合も、特定のセキュリティおよび管理上の目標に焦点を当てています。

ロールを作成してONTAPユーザアカウントにロールを割り当てる前に、次に示す主なセキュリティ要件とオプションを確認して準備しておく必要があります。また、次のWebサイトで一般的なワークフローの概念を確認してください。 "ワークフローを使用する準備をします"。

どの**ONTAP** リリースを使用していますか？

ONTAP リリースによって、使用可能なRESTエンドポイントとRBAC機能が決まります。

保護対象のリソースと範囲を特定

保護対象のリソースまたはコマンドとその範囲（クラスタまたはSVM）を特定する必要があります。

ユーザにはどのようなアクセス権が必要ですか。

リソースと範囲を特定したら、許可するアクセスレベルを決定する必要があります。

ユーザはどのように**ONTAP** にアクセスしますか。

ユーザはREST API、CLI、またはその両方を使用してONTAP にアクセスできます。

組み込みの役割の**1**つで十分か、またはカスタムの役割が必要か。

既存の組み込みロールを使用する方が便利ですが、必要に応じて新しいカスタムロールを作成することもできます。

どのような種類の役割が必要ですか？

セキュリティ要件とONTAP アクセスに基づいて、RESTと従来のどちらのロールを作成するかを選択する必要があります。

ロールの作成

SVMボリュームへのアクセスを制限する処理

SVM内でのストレージボリュームの管理を制限するロールを定義できます。

このワークフローについて

最初に、クローニングを除くすべての主要なボリューム管理機能へのアクセスを許可するために、トラディショナルロールが作成されます。ロールは次の特性で定義されます。

- GET、CREATE、MODIFY、DELETEなどのCRUDボリューム操作をすべて実行できる
- ボリュームクローンを作成できません

その後、必要に応じてロールを更新できます。このワークフローでは、2番目の手順でロールが変更され、ユ

ーザがボリュームクローンを作成できるようになります。

手順**1**：ロールを作成する

API呼び出しを問題してRBACロールを作成できます。

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
投稿（ Post ）	/api/security/rolesのように入力します

カールの例

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON の入力例

```
{
  "name": "role1",
  "owner": {
    "name": "cluster-1",
    "uuid": "852d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      { "path": "volume create", "access": "all" },
      { "path": "volume delete", "access": "all" }
    ]
}
```

手順**2**：ロールを更新する

API呼び出しを問題して既存のロールを更新できます。

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
投稿（ Post ）	/api/security/rolesのように入力します

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、ロールの定義が含まれているSVMのUUIDです。
$ロール名	パス	はい。	更新するSVM内のロールの名前を指定します。

カールの例

```
curl --request POST \
--location
"https://$FQDN_IP/api/security/roles/$SVM_ID/$ROLE_NAME/priveleges" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON の入力例

```
{
  "path": "volume clone",
  "access": "all"
}
```

データ保護の管理を実現

ユーザに提供できるデータ保護機能は限られています。

このワークフローについて

従来のロールは、次の特性で定義されます。

- Snapshotの作成と削除、およびSnapMirror関係の更新が可能です
- ボリュームやSVMなどの上位のオブジェクトを作成または変更することはできません

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/security/rolesのように入力します

カールの例

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON の入力例

```
{
  "name": "role1",
  "owner": {
    "name": "cluster-1",
    "uuid": "852d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      {"path": "volume snapshot create", "access": "all"},
      {"path": "volume snapshot delete", "access": "all"},
      {"path": "volume show", "access": "readonly"},
      {"path": "vserver show", "access": "readonly"},
      {"path": "snapmirror show", "access": "readonly"},
      {"path": "snapmirror update", "access": "all"}
  ]
}
```

ONTAPレポートの生成を許可する

ONTAP レポートを生成する機能をユーザに提供するRESTロールを作成できます。

このワークフローについて

作成されるロールは、次の特性で定義されます。

- 容量とパフォーマンス（ボリューム、qtree、LUN、アグリゲート、ノード、 SnapMirror関係の場合）
- 上位のオブジェクト（ボリュームやSVMなど）を作成または変更できない

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/security/rolesのように入力します

カールの例

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON の入力例

```
{
  "name": "rest_role1",
  "owner": {
    "name": "cluster-1",
    "uuid": "852d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      {"path": "/api/storage/volumes", "access": "readonly"},
      {"path": "/api/storage/qtrees", "access": "readonly"},
      {"path": "/api/storage/luns", "access": "readonly"},
      {"path": "/api/storage/aggregates", "access": "readonly"},
      {"path": "/api/cluster/nodes", "access": "readonly"},
      {"path": "/api/snapmirror/relationships", "access": "readonly"},
      {"path": "/api/svm/svms", "access": "readonly"}
  ]
}
```

ロールを持つユーザを作成します

このワークフローを使用して、RESTロールを関連付けたユーザを作成できます。

このワークフローについて

このワークフローには、カスタムRESTロールを作成して新しいユーザアカウントに関連付けるために必要な一般的な手順が含まれています。ユーザとロールの両方にSVMスコープがあり、特定のデータSVMに関連付けられています。一部の手順はオプションである場合もあれば、環境に応じて変更する必要がある場合もあります。

手順**1**：クラスタ内のデータ**SVM**をリストする

次のREST API呼び出しを実行して、クラスタ内のSVMを表示します。各SVMのUUIDと名前が出力に表示されます。

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/SVM/SVMs

カールの例

```
curl --request GET \
--location "https://$FQDN_IP/api/svm/svms?order_by=name" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

完了後

リストから必要なSVMを選択し、新しいユーザとロールを作成します。

手順2：SVMに定義されているユーザを表示する

選択したSVMで定義されているユーザを表示するために、次のREST API呼び出しを実行します。SVMは、ownerパラメータを使用して識別できます。

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/security/accounts（/api/security/アカウント）

カールの例

```
curl --request GET \
--location "https://$FQDN_IP/api/security/accounts?owner.name=dmp" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

完了後

SVMですでに定義されているユーザに基づいて、新しいユーザの一意の名前を選択します。

手順3：SVMに定義されているRESTロールを表示する

次のREST API呼び出しを実行し、選択したSVMで定義されているロールをリストします。SVMは、ownerパラメータを使用して識別できます。

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/security/rolesのように入力します

カールの例

```
curl --request GET \
--location "https://$FQDN_IP/api/security/roles?owner.name=dmp" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

完了後

SVMですでに定義されているロールに基づいて、新しいロールに一意の名前を選択します。

手順4：カスタムRESTロールを作成する

次のREST API呼び出しを実行して、SVMでカスタムのRESTロールを作成します。最初は権限を1つしか持たず、すべてのアクセスが拒否されるようにするために、このロールにはデフォルトのアクセスである* none *が設定されます。

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
投稿（ Post ）	/api/security/rolesのように入力します

カールの例

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON の入力例

```
{
  "name": "dprole1",
  "owner": {
    "name": "dmp",
    "uuid": "752d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      {"path": "/api", "access": "none"},
  ]
}
```

完了後

必要に応じて、手順3をもう一度実行して新しいロールを表示します。ONTAP CLIでもロールを表示できます。

手順**5**：権限を追加してロールを更新する

必要に応じて権限を追加してロールを変更するには、次のREST API呼び出しを実行します。

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
投稿（ Post ）	/api/security/roles/｛owner.uuid｝/｛name｝/privileges

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。	ロールの定義が含まれているSVMのUUID。
$ロール名	パス	はい。	更新するSVM内のロールの名前を指定します。

カールの例

```
curl --request POST \
--location
"https://$FQDN_IP/api/security/roles/$SVM_ID/$ROLE_NAME/privileges" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON の入力例

```
{
  "path": "/api/storage/volumes",
  "access": "readonly"
}
```

完了後

必要に応じて、手順3をもう一度実行して新しいロールを表示します。ONTAP CLIでもロールを表示できます。

手順6：ユーザを作成する

ユーザアカウントを作成するには、次のREST API呼び出しを実行します。上で作成したロール*dprole1*は、新しいユーザに関連付けられています。

ロールを指定せずにユーザを作成できます。この場合、ユーザにはデフォルトのロール（ `admin` または `vsadmin`）ユーザがクラスタスコープとSVMスコープのどちらで定義されているかに応じて変わります。別のロールを割り当てるには、ユーザを変更する必要があります。

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
投稿（ Post ）	/api/security/accounts（/api/security/アカウント）

カールの例

```
curl --request POST \
--location "https://$FQDN_IP/api/security/accounts" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON の入力例

```
{
  "owner": {"uuid":"daf84055-248f-11ed-a23d-005056ac4fe6"},
  "name": "david",
  "applications": [
      {"application":"ssh",
       "authentication_methods":["password"],
       "second_authentication_method":"none"}
  ],
  "role":"dprole1",
  "password":"netapp123"
}
```

完了後

SVM管理インターフェイスにサインインするには、新しいユーザのクレデンシャルを使用します。

著作権に関する情報

商標に関する情報