属性ベースのアクセス制御
ONTAP Technical Reports

NetApp
February 23, 2026

目次

属性ベースのアクセス制御

ONTAPによる属性ベースのアクセス制御

9.12.1以降では、NFSv4.2セキュリティラベルおよび拡張属性（xattrs）を使用してONTAPを設定し、属性および属性ベースアクセス制御（ABAC）を使用したロールベースアクセス制御（RBAC）をサポートできます。

ABACは、ユーザ属性、リソース属性、および環境条件に基づいて権限を定義する認可戦略です。ONTAPとNFS v4.2セキュリティラベルおよびxattrsの統合は、NIST Special Publication 800-162に規定されているABACソリューションのNIST標準に準拠しています。

NFS v4.2セキュリティラベルとxattrsを使用して、ファイルにユーザ定義の属性とラベルを割り当てることができます。ONTAPは、ABAC指向のIDおよびアクセス管理ソフトウェアと統合して、これらの属性とラベルに基づいてきめ細かなファイルおよびフォルダのアクセス制御ポリシーを適用できます。

関連情報

- "ONTAPを使用したABACへのアプローチ"
- "NetApp ONTAPのNFS：ベストプラクティスおよび実装ガイド"

ONTAPでの属性ベースアクセス制御（ABAC）のアプローチ

ONTAPには、NFS v4.2セキュリティラベルやNFSを使用した拡張属性（xattrs）など、ファイルレベルの属性ベースアクセス制御（ABAC）を実現するために使用できるいくつかのアプローチが用意されています。

NFS v4.2セキュリティラベル

ONTAP 9.9.1以降では、NFS v4.2の「ラベル付きNFS」機能がサポートされます。

NFS v4.2セキュリティラベルは、SELinuxラベルとMandatory Access Control（MAC；強制アクセス制御）を使用して、ファイルやフォルダへのきめ細かなアクセスを管理する方法です。これらのMACラベルはファイルとフォルダに格納され、UNIX権限およびNFS v4.x ACLと連携して機能します。

NFS v4.2セキュリティラベルがサポートされたことで、ONTAPはNFSクライアントのSELinuxラベル設定を認識して認識できるようになりました。NFS v4.2セキュリティラベルはRFC-7204で規定されています。

NFS v4.2セキュリティラベルのユースケースには、次のようなものがあります。

- 仮想マシン（VM）イメージのMACラベル付け
- 公共機関のデータセキュリティ分類（シークレット、トップシークレット、その他の分類）
- セキュリティコンプライアンス
- ディスクレス Linux

NFS v4.2セキュリティラベルを有効にする

NFS v4.2セキュリティラベルを有効または無効にするには、次のコマンドを使用します（advanced権限が必要）。

```
vserver nfs modify -vserver <svm_name> -v4.2-seclabel <disabled|enabled>
```

の詳細については `vserver nfs modify`、を"ONTAPコマンド リファレンス"参照してください。

NFS v4.2セキュリティラベルの適用モード

ONTAP 9.9.1以降では、ONTAPで次の強制モードがサポートされています。

- 制限付きサーバーモード：ONTAPはラベルを強制できませんが、ラベルを保存および送信できます。

  MACラベルを変更する機能は、強制するクライアントによって異なります。

- ゲストモード：クライアントにNFS対応（v4.1以前）のラベルが付けられていない場合、MACラベルは送信されません。

  ONTAPは現在、フルモード（MACラベルの保存と適用）をサポートしていません。

NFS v4.2セキュリティラベルの例

次の設定例は、Red Hat Enterprise Linuxリリース9.3（Plow）を使用した概念を示しています。

このユーザ `jrsmith`は、John R. Smithのクレデンシャルに基づいて作成され、次のアカウントPrivilegesを持ちます。

- ユーザ名= `jrsmith`
- Privileges = `uid=1112(jrsmith) gid=1112(jrsmith) groups=1112(jrsmith) context=user_u:user_r:user_t:s0`

ロールには2つあります。管理者アカウントは、次のMLS Privilegesの表で説明されているように、特権ユーザおよびユーザ `jrsmith`です。

ユーザ	ロール	タイプ	レベル
admins	sysadm_r	sysadm_t	t:s0
jrsmith	user_r	user_t	t:s1 - t:s4

この例の環境では、ユーザー `jrsmith`はに `s3`あるレベルのファイルにアクセスできます `s0`。以下に概説する既存のセキュリティ分類を強化して、管理者がユーザー固有のデータにアクセスできないようにすることができます。

- S0 =権限管理者ユーザデータ

- S0 =未分類データ
- S1 =社外秘
- S2 =シークレットデータ
- S3 =トップシークレットデータ

MCSヲシヨウシタ**NFS v4.2**セキュリティラベルノレイ

マルチレベルセキュリティ（MLS）に加えて、マルチカテゴリセキュリティ（MCS）と呼ばれる別の機能を使用すると、プロジェクトなどのカテゴリを定義できます。

NFSセキュリティラベル	値
`entitySecurityMark`	`t:s01 = UNCLASSIFIED`

拡張属性（**xattrs**）

ONTAP 9.12.1以降では、ONTAPはxattrs.xattrsをサポートしています。xattrsを使用すると、アクセス制御リスト(ACL)やユーザ定義属性など、システムによって提供されるもの以外のファイルやディレクトリにメタデータを関連付けることができます。

xattrsを実装するには、Linuxでと `getfattr`コマンドラインユーティリティを使用できます `setfattr`。これらのツールを使用すると、ファイルやディレクトリの追加メタデータを強力に管理できます。不適切な使用は予期しない動作やセキュリティ上の問題につながる可能性があるため、注意して使用する必要があります。使用方法の詳細については、および `getfattr`のマニュアルページを参照するか、信頼性の高いその他のドキュメントを参照して `setfattr`ください。

ONTAPファイルシステムでxattrsが有効になっている場合'ユーザーはファイルの任意の属性を設定'変更'取得できますこれらの属性は、アクセス制御情報など、標準のファイル属性セットではキャプチャされないファイルに関する追加情報を格納するために使用できます。

ONTAPでxattrsを使用するには、いくつかの要件と制限があります。

- Red Hat Enterprise Linux 8.4以降
- Ubuntu 22.04以降
- 各ファイルには最大128個のxattrsを含めることができます。
- xattrキーは255バイトに制限されています
- キーまたは値の合計サイズは'xattrごとに1,729バイトです
- ディレクトリとファイルはxattrsを持つことができる
- xattrsを設定および取得するには、ユーザおよびグループに対して書き込みモードビットが有効になっている必要があります。 `w`

Xattrsはユーザーネームスペース内で使用され、ONTAP自体に本質的な意味を持たない。代わりに、それらの実用的なアプリケーションは、ファイルシステムとやり取りするクライアント側のアプリケーションによって排他的に決定され、管理されます。

xattrの使用例：

- ファイルの作成を担当するアプリケーションの名前の記録
- ファイルの取得元の電子メールメッセージへの参照の維持
- ファイルオブジェクトを整理するための分類フレームワークの確立
- 元のダウンロード元のURLを使用したファイルのラベル付け

xattrsの管理用コマンド

- `setfattr`ファイルまたはディレクトリの拡張属性を設定します。

  ```
  setfattr -n <attribute_name> -v <attribute_value> <file or directory name>
  ```

  コマンド例：

  ```
  setfattr -n user.comment -v test example.txt
  ```

- `getfattr`特定の拡張属性の値を取得するか、ファイルまたはディレクトリのすべての拡張属性を一覧表示します。

  特定の属性： `getfattr -n <attribute_name> <file or directory name>`

  すべての属性： `getfattr <file or directory name>`

  コマンド例：

  ```
  getfattr -n user.comment example.txt
  ```

xattrキーと値のペアの例

次の表に、2つのxattrキー値ペアの例を示します。

xattr	値
`user.digitalIdentifier`	`CN=John Smith jrsmith, OU=Finance, OU=U.S.ACME, O=US, C=US`
`user.countryOfAffiliations`	`USA`

xattrsの**ACE**を使用したユーザー権限

Access Control Entry（ACE；アクセス制御エントリ）は、ファイルやディレクトリなどの特定のリソースに対して個 々 のユーザまたはユーザグループに付与されるアクセス権または権限を定義するACL内のコンポーネントです。各ACEは、許可または拒否されるアクセスのタイプを指定し、特定のセキュリティプリンシパル（ユーザまたはグループのID）に関連付けます。

xattrsに必要なアクセス制御エントリ（**ACE**）

- Retrieve xattr：ユーザがファイルまたはディレクトリの拡張属性を読み取るために必要な権限。「R」は、読み取り権限が必要であることを示します。
- set xattrs：拡張属性を変更または設定するために必要な権限。"A"、"w"、"T"は、append、write、xattrsに関連する特定のパーミッションなど、パーミッションの異なる例を表しています。
- ファイル:拡張属性を設定するには、追加、書き込み、およびxattrsに関連する特別な権限が必要です。
- ディレクトリ:拡張属性を設定するには、特定の権限「T」が必要です。

ファイルタイプ	**xattr**の取得	**xattrs**の設定
ファイル	R	A、w、T
ディレクトリ	R	T

ABAC IDおよびアクセス制御ソフトウェアとの統合

ABACの機能を最大限に活用するために、ONTAPはABAC指向のIDおよびアクセス管理ソフトウェアと統合できます。

ABACシステムでは、Policy Enforcement Point (PEP)とPolicy Decision Point (PDP)が重要な役割を果たす。PEPはアクセス制御ポリシーの適用を担当し、PDPはポリシーに基づいてアクセスを許可するか拒否するかを決定します。

実際的な設定では、NFSセキュリティラベルとxattrsを組み合わせて使用します。これらは、分類、セキュリティ、アプリケーション、コンテンツなど、さまざまなメタデータを表すために使用されます。これらはすべてABACの決定を行うのに役立ちます。xattrsは、PDPが意思決定プロセスに使用するリソース属性を格納するために使用できます。属性は、ファイルの分類レベルを表すように定義できます（「未分類」、「機密」、「シークレット」、「トップシークレット」など）。その後、PDPはこの属性を使用して、ユーザーがクリアランスレベル以下の分類レベルを持つファイルのみにアクセスするように制限するポリシーを適用できます。

このコンテンツでは、お客様のID、認証、およびアクセスサービスに、ファイルシステムへのアクセスの仲介者として機能するPEPおよびPDPが少なくとも1つ含まれていることを前提としています。

ABACのプロセスフローの例

1. ユーザは、PEPへのシステムアクセスにクレデンシャル（PKI、OAuth、SAMLなど）を提示し、PDPから結果を取得します。

   PEPの役割は、ユーザのアクセス要求を代行受信してPDPに転送することです。

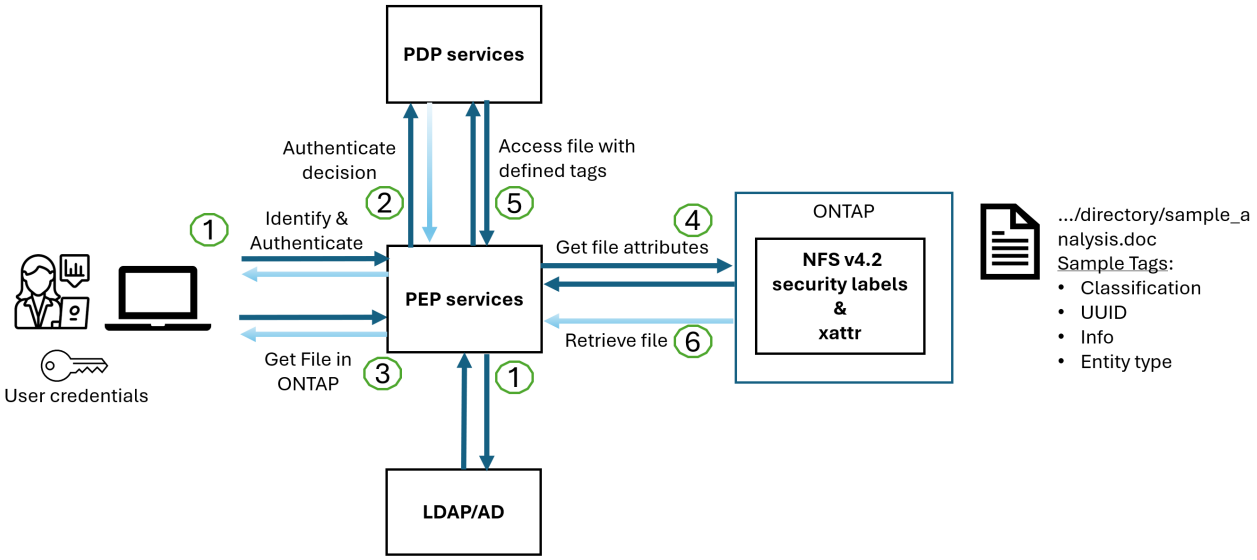
2. PDPは、確立されたABACポリシーに照らしてこの要求を評価します。

   これらのポリシーでは、ユーザー、問題のリソース、および周囲の環境に関連するさまざまな属性が考慮されます。これらのポリシーに基づいて、PDPはアクセスを許可するか拒否するかを決定し、その決定をPEPに伝えます。

   PDPはPEPにポリシーを提供して実施します。PEPはこの決定を実行し、PDPの決定に従ってユーザーのアクセス要求を許可または拒否します。

3. 要求が成功すると、ユーザはONTAPに格納されているファイル（AFF、AFF -Cなど）を要求します。
4. 要求が成功すると、PEPはドキュメントから詳細なアクセス制御タグを取得します。
5. PEPは、そのユーザの証明書に基づいてユーザのポリシーを要求します。
6. ユーザがファイルにアクセスできる場合、PEPはポリシーとタグに基づいて決定を行い、ユーザがファイルを取得できるようにします。

実際のアクセスはトークンを使用して行われる場合があります。



ONTAPクローニングとSnapMirror

ONTAPのクローニングおよびSnapMirrorテクノロジは、効率的で信頼性の高いデータレプリケーションおよびクローニング機能を提供するように設計されています。xattrsは、ファイルに関連付けられた追加のメタデータ（セキュリティラベル、アクセス制御情報、ユーザ定義データなど）を保存するため、ファイルのコンテキストと整合性の維持に不可欠です。xattrsは重要です。

ONTAPのFlexCloneテクノロジを使用してボリュームをクローニングすると、ボリュームの完全な書き込み可能なレプリカが作成されます。このクローニングプロセスは瞬時に実行されるスペース効率に優れており、すべてのファイルデータとメタデータが含まれているため、xattrsを完全にレプリケートできます。同様に、SnapMirrorでは、データが完全に忠実にセカンダリシステムにミラーリングされます。これにはxattrsも含まれます。xattrsは、このメタデータに依存するアプリケーションが正しく機能するために非常に重要です。

NetApp ONTAPでは、クローニング処理とレプリケーション処理の両方にxattrsを含めることで、プライマリストレージシステムとセカンダリストレージシステム全体で、すべての特性を含む完全なデータセットを使用して一貫性を確保します。この包括的なデータ管理アプローチは、一貫したデータ保護、迅速なリカバリ、コンプライアンスと規制基準への準拠を必要とする組織にとって不可欠です。また、オンプレミスでもクラウドでも、さまざまな環境にわたってデータの管理が簡易化されるため、ユーザはプロセス中もデータが完全で変更されていないという安心感を得ることができます。

NFS v4.2セキュリティラベルには、に定義された注意事項NFS v4.2セキュリティラベルがあります。

ラベルに対する変更の監査

xattrsまたはNFSセキュリティラベルに対する変更の監査は、ファイルシステムの管理とセキュリティの重要な側面です。標準のファイルシステム監査ツールを使用すると、xattrsやセキュリティラベルの変更など、ファイルシステムに対するすべての変更を監視およびロギングできます。

Linux環境では、 `auditd`ファイルシステムイベントの監査を確立するために一般にデーモンが使用されます。管理者は、`xattr`の変更（、 `lsetxattr`など）に関連する特定のシステムコールを監視し、 `fsetxattr`属性と、 `lremovexattr`fremovexattr`の設定、および`removexattr`属性の削除を監視するルールを設定でき`setxattr`ます。

ONTAP FPolicyは、ファイル操作をリアルタイムで監視および制御するための堅牢なフレームワークを提供することで、これらの機能を拡張します。FPolicyは、さまざまな属性xattrイベントをサポートするように設定できます。これにより、ファイル操作をきめ細かく制御したり、包括的なデータ管理ポリシーを適用したりできます。

xattrsを使用するユーザ、特にNFS v3およびNFS v4環境では、監視対象としてサポートされるファイル操作とフィルタの特定の組み合わせのみが対象となります。FPolicyによるNFS v3およびNFS v4のファイルアクセスイベントの監視でサポートされるファイル操作とフィルタの組み合わせを次に示します。

サポートされているファイル操作	サポートされているフィルタ
`setattr`	`offline-bit, setattr_with_owner_change, setattr_with_group_change, setattr_with_mode_change, setattr_with_modify_time_change, setattr_with_access_time_change, setattr_with_size_change, exclude_directory`

属性設定操作の**auditd**ログスニペットの例：

```
type=SYSCALL msg=audit(1713451401.168:106964): arch=c000003e syscall=188
success=yes exit=0 a0=7fac252f0590 a1=7fac251d4750 a2=7fac252e50a0 a3=25
items=1 ppid=247417 pid=247563 auid=1112 uid=1112 gid=1112 euid=1112
suid=1112 fsuid=1112 egid=1112 sgid=1112 fsgid=1112 tty=pts0 ses=141
comm="python3" exe="/usr/bin/python3.9"
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
key="*set-xattr*"ARCH=x86_64 SYSCALL=**setxattr** AUID="jrsmith"
UID="jrsmith" GID="jrsmith" EUID="jrsmith" SUID="jrsmith"
FSUID="jrsmith" EGID="jrsmith" SGID="jrsmith" FSGID="jrsmith"
```

ユーザがxattrsを使用できるようにする"ONTAP FPolicy"と、ファイルシステムの整合性とセキュリティを維持するために不可欠な可視性と制御のレイヤが提供されます。FPolicyの高度な監視機能を活用することで、組織はxattrsに対するすべての変更を追跡、監査し、セキュリティおよびコンプライアンス基準に準拠させることができます。ファイルシステム管理に対するこのプロアクティブなアプローチが、データガバナンスと保護戦略を強化したいと考えている組織にとって、ONTAP FPolicyを有効にすることが強く推奨される理由です。

データアクセスの制御例

John R. SmithのPKI証明書に格納されているデータの次のエントリ例は、NetAppのアプローチをファイルに適用し、きめ細かなアクセス制御を提供する方法を示しています。

これらの例は説明を目的としたものであり、NFS v4.2セキュリティラベルおよびxattrsに関連付けられているメタデータはお客様の責任で確認してください。わかりやすいように更新とラベルの保持の詳細は省略しています。

- PKI証明書値の例*

キー	値
entitySecurityMark	T：S01 =未分類

情報	{
	"commonName": {
	"value": "Smith John R jrsmith"
	},
	"emailAddresses": [
	{
	"value": "jrsmith@dod.mil"
	}
	],
	"employeeId": {
	"value": "00000387835"
	},
	"firstName": {
	"value": "John"
	},
	"lastName": {
	"value": "Smith"
	},
	"telephoneNumber": {
	"value": "938/260-9537"
	},
	"uid": {
	"value": "jrsmith"
	}
	}

仕様	"DoD"
UUID	b4111349-7875-4115-AD30-0928565f2e15

キー	値

管理組織	{
	"value": "DoD"
	}

ブリーフィング	[
	{
	"value": "ABC1000"
	},
	{
	"value": "DEF1001"
	},
	{
	"value": "EFG2000"
	}
	]

市民権ステータス	{
	"value": "US"
	}

クリアランス	[
	{
	"value": "TS"
	},
	{
	"value": "S"
	},
	{
	"value": "C"
	},
	{
	"value": "U"
	}
	]

キー	値

加盟国	<pre>[
	  {
	  "value": "USA"
	  }
	]</pre>

デジタル識別子	<pre>{
	  "classification": "UNCLASSIFIED",
	  "value": "cn=smith john r jrsmith, ou=dod, o=u.s.
	government, c=us"
	}</pre>

転送先	<pre>{
	  "value": "DoD"
	}</pre>

DutyOrganization	<pre>{
	  "value": "DoD"
	}</pre>

エンティティタイプ	<pre>{
	  "value": "GOV"
	}</pre>

キー	値

FineAccessControls	[
	{
	"value": "SI"
	},
	{
	"value": "TK"
	},
	{
	"value": "NSYS"
	}
	]

これらのPKIエンタイトルメントには、データ型やアトリビューションによるアクセスなど、John R. Smithのアクセスの詳細が表示されます。

IC-TDFメタデータがファイルとは別に格納されているシナリオでは、NetAppは詳細なアクセス制御レイヤを追加することを推奨しています。これには、アクセス制御情報がディレクトリレベルおよび各ファイルに関連付けられて格納されることが含まれます。例として、次のタグがファイルにリンクされているとします。

- NFS v4.2セキュリティラベル：セキュリティの決定に使用
- xattrs：ファイルおよび組織のプログラム要件に関連する補足情報を提供します。

次のキーと値のペアは、xattrsとして保存できるメタデータの例であり、ファイルの作成者と関連するセキュリティ分類に関する詳細情報を提供します。クライアントアプリケーションでこのメタデータを使用すると、十分な情報に基づいてアクセスに関する意思決定を行い、組織の標準や要件に従ってファイルを整理できます。

- xattrキーと値のペアの例*

キー	値
`user.uuid`	`"761d2e3c-e778-4ee4-997b-3bb9a6a1d3fa"`
`user.entitySecurityMark`	`"UNCLASSIFIED"`
`user.specification`	`"INFO"`

キー	値

user.Info	{
	"commonName": {
	"value": "Smith John R jrsmith"
	},
	"currentOrganization": {
	"value": "TUV33"
	},
	"displayName": {
	"value": "John Smith"
	},
	"emailAddresses": [
	"jrsmith@example.org"
	],
	"employeeId": {
	"value": "00000405732"
	},
	"firstName": {
	"value": "John"
	},
	"lastName": {
	"value": "Smith"
	},
	"managers": [
	{
	"value": ""
	}
	],
	"organizations": [
	{
	"value": "TUV33"
	},
	{
	"value": "WXY44"
	}
	],
	"personalTitle": {
	"value": ""
	},
	"secureTelephoneNumber": {
	"value": "506-7718"
	},
	"telephoneNumber": {
	"value": "264/160-7187"
	},
	"title": {
	"value": "Software Engineer"
	},

キー	値
user.geo_point	[-78.7941, 35.7956]

```
    }
  }
```

関連情報

- "NetApp ONTAPのNFS：ベストプラクティスおよび実装ガイド"
- "ONTAPコマンド リファレンス"
- コメント要求（RFC）
  - "RFC 7204:ラベル付きNFSの要件"
  - "RFC 2203：RPCSEC_GSS Protocol Specification"
  - "RFC 3530：Network File System (NFS) Version 4 Protocol"

著作権に関する情報

商標に関する情報