Attributbasierte Zugriffssteuerung

ONTAP Technical Reports

NetApp
January 23, 2026

Inhalt

Attributbasierte Zugriffssteuerung

Attributbasierte Zugriffssteuerung mit ONTAP

Ab Version 9.12.1 können Sie ONTAP mit NFSv4.2-Sicherheitsetiketten und erweiterten Attributen (xattrs) konfigurieren, um rollenbasierte Zugriffssteuerung (Role Based Access Control, RBAC) mit Attributen und attributbasierter Zugriffssteuerung (Attribute Based Access Control, ABAC) zu unterstützen.

ABAC ist eine Autorisierungsstrategie, die Berechtigungen basierend auf Benutzerattributen, Ressourcenattributen und Umgebungsbedingungen definiert. Die Integration von ONTAP mit NFS v4.2 Security Labels und xattrs entspricht den NIST Standards für ABAC Lösungen, wie in NIST Special Publication 800-162.

Sie können NFS v4.2-Sicherheitsetiketten und xattrs verwenden, um Dateien benutzerdefinierte Attribute und Labels zuzuweisen. ONTAP kann in die ABAC-orientierte Identitäts- und Zugriffsmanagement-Software integriert werden, um auf der Grundlage dieser Attribute und Labels granulare Richtlinien zur Zugriffskontrolle von Dateien und Ordnern durchzusetzen.

Verwandte Informationen

- "Ansätze für ABAC mit ONTAP"
- "NFS in NetApp ONTAP: Best Practice und Implementierungsleitfaden"

Ansätze zur attributbasierten Zugriffssteuerung (ABAC) in ONTAP

ONTAP bietet verschiedene Ansätze zur Erzielung einer attributbasierten Zugriffssteuerung (File-Level-Based Access Control, ABAC), einschließlich NFS v4.2 Security Labels und Extended Attributes (xattrs) mithilfe von NFS.

NFS v4.2-Sicherheitslabels

Ab ONTAP 9.9 wird die NFS v4.2-Funktion mit der Bezeichnung NFS unterstützt.

NFS v4.2-Sicherheitsetiketten sind eine Möglichkeit, den granularen Datei- und Ordnerzugriff mithilfe von SELinux-Labels und Mandatory Access Control (MAC) zu verwalten. Diese MAC-Labels werden mit Dateien und Ordnern gespeichert und funktionieren in Verbindung mit UNIX-Berechtigungen und NFS v4.x ACLs.

Durch die Unterstützung von NFS v4.2-Sicherheitsetiketten erkennt ONTAP jetzt die SELinux-Label-Einstellungen des NFS-Clients und versteht sie. Die Sicherheitslabels für NFS v4.2 sind in RFC-7204 abgedeckt.

Zu den Anwendungsfällen für die NFS v4.2-Sicherheitslabels gehören:

- MAC-Beschriftung von Virtual Machine (VM) Images
- Datensicherheitsklassifizierung für den öffentlichen Sektor (geheime, streng geheime und andere Klassifizierungen)
- Sicherheits-Compliance

- Diskless Linux

Aktivieren Sie die NFS v4.2-Sicherheitsetiketten

Sie können die NFS v4.2-Sicherheitsetiketten mit dem folgenden Befehl aktivieren oder deaktivieren (erweiterte Berechtigung erforderlich):

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

Erfahren Sie mehr über `vserver nfs modify` in der "ONTAP-Befehlsreferenz".

Durchsetzungsmodi für NFS v4.2-Sicherheitslabels

Ab ONTAP 9.9 unterstützt ONTAP die folgenden Erzwingungsmodi:

- **Eingeschränkter Servermodus**: ONTAP kann die Labels nicht erzwingen, sondern speichern und übertragen.

  Die Möglichkeit, MAC-Labels zu ändern, liegt bei der Durchsetzung durch den Client.

- **Gastmodus**: Wenn der Client nicht NFS-aware (v4.1 oder niedriger) ist, werden MAC-Labels nicht übertragen.

  ONTAP unterstützt derzeit nicht den Vollmodus (Speichern und Erzwingen von MAC-Etiketten).

Beispiele für Sicherheitsetiketten in NFS v4.2

Die folgende Beispielkonfiguration zeigt Konzepte mit Red hat Enterprise Linux Version 9.3 (Plough).

Der Benutzer `jrsmith`, der basierend auf den Anmeldeinformationen von John R. Smith erstellt wurde, hat das folgende Konto Privileges:

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

Es gibt zwei Rollen: Das Administratorkonto, das ein privilegierter Benutzer und ein Benutzer ist `jrsmith`, wie in der folgenden MLS-Privileges-Tabelle beschrieben:

Benutzer	Rolle	Typ	Stufen
`admins`	`sysadm_r`	`sysadm_t`	`t:s0`
`jrsmith`	`user_r`	`user_t`	`t:s1 - t:s4`

In dieser Beispielumgebung hat der Benutzer `jrsmith` Zugriff auf Dateien auf den Ebenen `s0` bis `s3`. Wir können die bestehenden Sicherheitsklassifizierungen wie unten beschrieben verbessern, um sicherzustellen, dass Administratoren keinen Zugriff auf benutzerspezifische Daten haben.

- s0 = Berechtigungsverwaltung Benutzerdaten
- s0 = nicht klassifizierte Daten
- s1 = vertraulich
- s2 = geheime Daten
- s3 = Top-Geheimdaten

Beispiel für NFS v4.2-Sicherheitsetiketten mit MCS

Zusätzlich zu Multi-Level Security (MLS) können Sie mit einer weiteren Funktion namens Multi-Category Security (MCS) Kategorien wie Projekte definieren.

NFS-Sicherheitsetikett	Wert
`entitySecurityMark`	`t:s01 = UNCLASSIFIED`

Erweiterte Attribute (xattrs)

Ab ONTAP 9.12.1 unterstützt ONTAP xattrs. Xattrs ermöglicht die Zuordnung von Metadaten zu Dateien und Verzeichnissen über das hinaus, was vom System bereitgestellt wird, wie z. B. Zugriffskontrolllisten (ACLs) oder benutzerdefinierte Attribute.

Um xattrs zu implementieren, können Sie und `getfattr` Kommandozeilen-Dienstprogramme in Linux verwenden `setfattr`. Diese Tools bieten eine leistungsstarke Möglichkeit, zusätzliche Metadaten für Dateien und Verzeichnisse zu managen. Sie sollten mit Vorsicht eingesetzt werden, da eine unsachgemäße Verwendung zu unerwartetem Verhalten oder Sicherheitsproblemen führen kann. Detaillierte Anweisungen zur Verwendung finden Sie stets auf den `setfattr` Manpages und `getfattr` in anderen zuverlässigen Dokumentationen.

Wenn xattrs auf einem ONTAP-Dateisystem aktiviert ist, können Benutzer beliebige Attribute auf Dateien festlegen, ändern und abrufen. Diese Attribute können verwendet werden, um zusätzliche Informationen über die Datei zu speichern, die nicht von den standardmäßigen Dateiattributen erfasst werden, z. B. Informationen zur Zugriffssteuerung.

Für die Verwendung von xattrs in ONTAP gibt es mehrere Anforderungen und Grenzen:

- Red hat Enterprise Linux 8.4 oder höher
- Ubuntu 22.04 oder höher
- Jede Datei kann bis zu 128 xattrs haben
- Xattr-Schlüssel sind auf 255 Byte begrenzt
- Die kombinierte Schlüssel- oder Wertgröße beträgt 1,729 Byte pro xattr
- Verzeichnisse und Dateien können xattrs haben
- Zum Festlegen und Abrufen von xattrs `w` oder Schreibmodus müssen Bits für den Benutzer und die Gruppe aktiviert sein

Xattrs werden innerhalb des Benutzer-Namespace verwendet und haben keine intrinsische Bedeutung für ONTAP selbst. Stattdessen werden ihre praktischen Anwendungen ausschließlich von der Client-seitigen Anwendung bestimmt und verwaltet, die mit dem Dateisystem interagiert.

Anwendungsbeispiele für xattr:

- Aufzeichnen des Namens der Anwendung, die für die Erstellung einer Datei verantwortlich ist
- Beibehalten eines Verweises auf die E-Mail-Nachricht, aus der eine Datei abgerufen wurde
- Einrichten eines Kategorisierungsrahmens für die Organisation von Dateiobjekten
- Beschriften von Dateien mit der URL ihrer ursprünglichen Download-Quelle

Befehle zum Verwalten von xattrs

- `setfattr` Legt ein erweitertes Attribut einer Datei oder eines Verzeichnisses fest:

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

  Beispielbefehl:

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

- `getfattr` Ruft den Wert eines bestimmten erweiterten Attributs ab oder listet alle erweiterten Attribute einer Datei oder eines Verzeichnisses auf:

  Spezifisches Attribut: `getfattr -n <attribute_name> <file or directory name>`

  Alle Attribute: `getfattr <file or directory name>`

  Beispielbefehl:

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

Beispiele für das Schlüsselwertpaar xattr

In der folgenden Tabelle sind zwei Beispiele für das Schlüsselwertpaar xattr aufgeführt:

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

Benutzerberechtigungen mit ACE für xattrs

Ein Access Control Entry (ACE) ist eine Komponente innerhalb einer ACL, die die Zugriffsrechte oder Berechtigungen definiert, die einem einzelnen Benutzer oder einer Benutzergruppe für eine bestimmte Ressource, z. B. eine Datei oder ein Verzeichnis, gewährt werden. Jeder ACE gibt die Art des erlaubten oder abgelehnten Zugriffs an und ist mit einem bestimmten Sicherheitsprinzipal (Benutzer- oder Gruppenidentität) verknüpft.

Access Control Entry (ACE) für xattrs erforderlich

- Abrufen von xattr: Die Berechtigungen, die ein Benutzer benötigt, um die erweiterten Attribute einer Datei oder eines Verzeichnisses zu lesen. Das „R“ bedeutet, dass Leseberechtigung erforderlich ist.
- Xattrs festlegen: Die Berechtigungen, die zum Ändern oder Festlegen der erweiterten Attribute benötigt werden. „A“, „w“ und „T“ stellen verschiedene Beispiele für Berechtigungen wie Append, Write und eine bestimmte Berechtigung in Bezug auf xattrs dar.
- Dateien: Benutzer benötigen Append, Write und möglicherweise eine spezielle Berechtigung im Zusammenhang mit xattrs, um erweiterte Attribute zu setzen.
- Verzeichnisse: Eine bestimmte Berechtigung "T" ist erforderlich, um erweiterte Attribute zu setzen.

Dateityp	Xattr. Abrufen	Xattrs einstellen
Datei	R	A,w,T
Verzeichnis	R	T

Integration mit ABAC Identitäts- und Zugriffskontrollsoftware

Um die Funktionen von ABAC voll auszuschöpfen, kann ONTAP in eine ABAC-orientierte Identitäts- und Zugriffsverwaltungssoftware integriert werden.

In einem ABAC-System spielen der Policy Enforcement Point (PEP) und der Policy Decision Point (PDP) eine entscheidende Rolle. Der PEP ist für die Durchsetzung von Zugriffssteuerungsrichtlinien verantwortlich, während der PDP die Entscheidung darüber trifft, ob der Zugriff auf der Grundlage der Richtlinien gewährt oder verweigert werden soll.

In einer praktischen Umgebung würde ein Unternehmen eine Mischung aus NFS-Sicherheitsetiketten und xattrs einsetzen. Diese werden verwendet, um eine Vielzahl von Metadaten darzustellen, einschließlich Klassifizierung, Sicherheit, Anwendung und Inhalt, die alle entscheidend für ABAC Entscheidungen sind. Xattrs, zum Beispiel, kann verwendet werden, um die Ressourcenattribute zu speichern, die die PDP für seinen Entscheidungsprozess verwendet. Ein Attribut kann definiert werden, um die Klassifizierungsstufe einer Datei darzustellen (z. B. „nicht klassifiziert“, „vertraulich“, „geheim“ oder „streng geheim“). Die PDP könnte dann dieses Attribut nutzen, um eine Richtlinie durchzusetzen, die Benutzern den Zugriff auf Dateien einschränkt, die eine Klassifizierungsstufe haben, die ihrem Sicherheitsniveau entspricht oder kleiner ist.

Dieser Inhalt setzt voraus, dass die Identitäts-, Authentifizierungs- und Zugriffsdienste des Kunden mindestens einen PEP und ein PDP umfassen, die als Vermittler für den Zugriff auf das Dateisystem fungieren.

Beispiel für einen Prozessablauf für ABAC

1. Benutzer stellt Anmeldeinformationen (z. B. PKI, OAuth, SAML) für den Systemzugriff auf PEP bereit und ruft Ergebnisse von PDP ab.

   Die Rolle des PEP besteht darin, die Zugriffsanforderung des Benutzers abzufangen und an das PDP weiterzuleiten.

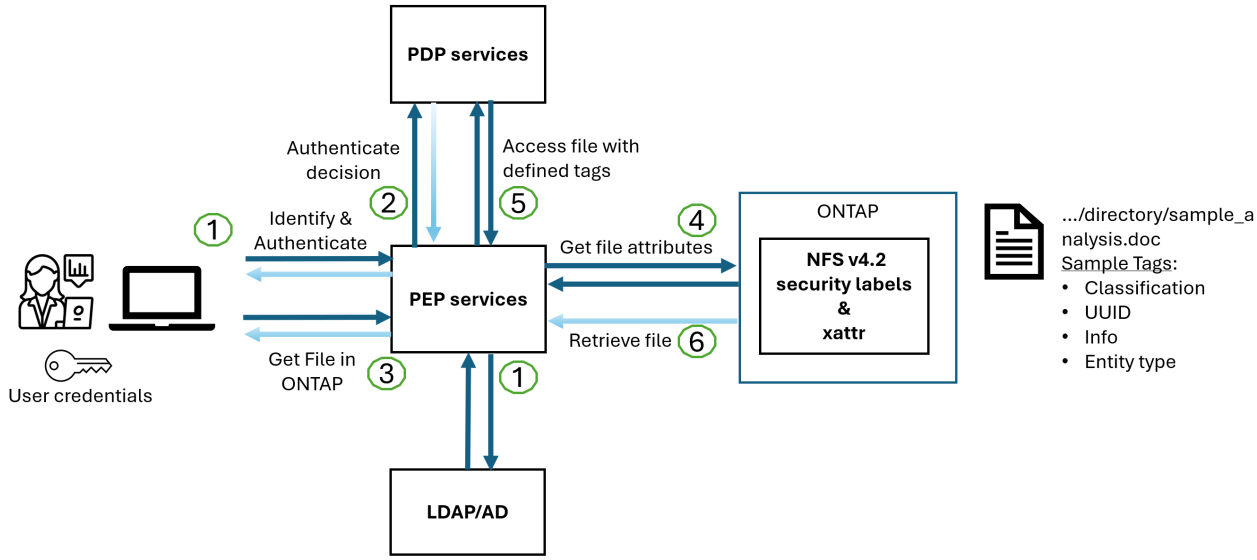
2. Die PDP wertet diese Anforderung dann anhand der festgelegten ABAC-Richtlinien aus.

   In diesen Richtlinien werden verschiedene Attribute berücksichtigt, die sich auf den Benutzer, die betreffende Ressource und die Umgebung beziehen. Auf der Grundlage dieser Richtlinien trifft die PDP eine Zugriffsentscheidung, entweder zuzulassen oder abzulehnen, und teilt diese Entscheidung dann dem PEP zurück.

PDP stellt PEP Richtlinien zur Durchsetzung bereit. Der PEP erzwingt dann diese Entscheidung, indem er die Zugriffsanfrage des Benutzers gemäß der Entscheidung des PDP entweder gewährt oder ablehnt.

3. Nach einer erfolgreichen Anfrage fordert der Benutzer eine in ONTAP gespeicherte Datei an (z. B. AFF, AFF-C).
4. Wenn die Anforderung erfolgreich war, erhält PEP fein abgestufte Zugangskontroll-Tags aus dem Dokument.
5. PEP fordert die Richtlinie für den Benutzer auf Grundlage der Zertifikate dieses Benutzers an.
6. PEP trifft eine Entscheidung auf der Grundlage von Richtlinien und Tags, wenn der Benutzer Zugriff auf die Datei hat, und lässt den Benutzer die Datei abrufen.

Der eigentliche Zugriff kann mit Token erfolgen.



ONTAP Cloning und SnapMirror

Die Klon- und SnapMirror-Technologien von ONTAP bieten effiziente und zuverlässige Datenreplizierungs- und Klonfunktionen und stellen sicher, dass alle Aspekte von Dateidaten, einschließlich xattrs, zusammen mit der Datei erhalten und übertragen werden. Xattrs sind wichtig, da sie zusätzliche Metadaten, die einer Datei zugeordnet sind, wie z. B. Sicherheitslabels, Zugriffskontrollinformationen und benutzerdefinierte Daten, speichern. Diese sind für die Aufrechterhaltung des Kontexts und der Integrität dieser Datei von wesentlicher Bedeutung.

Wenn ein Volume mit der FlexClone-Technologie von ONTAP geklont wird, wird ein exaktes, beschreibbares Replikat des Volumes erstellt. Dieser Klonprozess ist sofort und platzsparend und umfasst alle Dateidaten und Metadaten, um sicherzustellen, dass xattrs vollständig repliziert werden. SnapMirror sorgt auf ähnliche Weise dafür, dass Daten originalgetreu auf ein sekundäres System gespiegelt werden. Dazu gehört xattrs, die entscheidend sind für Anwendungen, die auf diese Metadaten angewiesen sind, um korrekt zu funktionieren.

Durch die Einbeziehung von xattrs sowohl beim Klonen als auch bei der Replizierung stellt NetApp ONTAP sicher, dass der vollständige Datensatz mit allen seinen Merkmalen verfügbar und konsistent über primäre und sekundäre Storage-Systeme hinweg ist. Dieser umfassende Datenmanagementansatz ist für Unternehmen unerlässlich, die eine konsistente Datensicherung, schnelle Wiederherstellung und die Einhaltung von Compliance- und gesetzlichen Standards benötigen. Zudem vereinfacht sie das Management von Daten in

verschiedenen Umgebungen, sowohl vor Ort als auch in der Cloud. Benutzer können sich darauf verlassen, dass ihre Daten während dieser Prozesse vollständig und unverändert sind.

Für NFS v4.2-Sicherheits-Labels sind die Einschränkungen definiert inNFS v4.2-Sicherheitslabels.

Prüfen von Änderungen an Beschriftungen

Das Auditing von Änderungen an xattrs oder NFS-Sicherheitsetiketten ist ein wichtiger Aspekt der Verwaltung und Sicherheit von Dateisystemen. Standard-Dateisystemauditing-Tools ermöglichen die Überwachung und Protokollierung aller Änderungen an einem Dateisystem, einschließlich Änderungen an xattrs und Sicherheitsetiketten.

In Linux-Umgebungen wird der `auditd` Daemon häufig verwendet, um Auditing für Dateisystemereignisse einzurichten. Es ermöglicht Administratoren, Regeln zu konfigurieren, um auf bestimmte Systemaufrufe im Zusammenhang mit xattr-Änderungen zu achten, wie `setxattr`,, `lsetxattr` und `fsetxattr` um Attribute und, `lremovexattr` zu setzen `removexattr` und `fremovexattr` Attribute zu entfernen.

ONTAP FPolicy erweitert diese Funktionen durch ein robustes Framework für das Monitoring und die Kontrolle von Dateivorgängen in Echtzeit. FPolicy kann zur Unterstützung verschiedener xattr-Ereignisse konfiguriert werden. Dies ermöglicht eine granulare Kontrolle über Dateivorgänge und die Durchsetzung umfassender Datenmanagement-Richtlinien.

Für Benutzer, die xattrs verwenden, insbesondere in NFS v3- und NFS v4-Umgebungen, werden nur bestimmte Kombinationen von Dateioperationen und -Filtern für die Überwachung unterstützt. Die Liste der unterstützten Dateioperationen und Filterkombinationen für das FPolicy Monitoring von NFS v3- und NFS v4-Dateizugriffsereignissen ist unten detailliert:

Unterstützte Dateivorgänge	Unterstützte Filter
`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`

Beispiel eines auditd-Protokollausschlags für eine setattr-Operation:

```
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"
```

Die Aktivierung "ONTAP FPolicy" für Benutzer, die mit xattrs arbeiten, bietet eine Ebene der Sichtbarkeit und

Kontrolle, die für die Aufrechterhaltung der Integrität und Sicherheit des Dateisystems unerlässlich ist. Mithilfe der erweiterten Monitoring-Funktionen von FPolicy können Unternehmen sicherstellen, dass alle Änderungen an xattrs nachverfolgt, geprüft und an ihren Sicherheits- und Compliance-Standards ausgerichtet werden. Dieser proaktive Ansatz beim Filesystem-Management ist daher die Aktivierung von ONTAP FPolicy nur für Unternehmen empfehlenswert, die ihre Daten-Governance- und Sicherungsstrategien verbessern möchten.

Beispiele für die Kontrolle des Zugriffs auf Daten

Der folgende Beispieleintrag für Daten, die in John R. Smiths PKI-Zertifikat gespeichert sind, zeigt, wie der Ansatz von NetApp auf eine Datei angewendet werden kann und eine feingranulare Zugriffskontrolle bietet.

Diese Beispiele dienen zur Veranschaulichung, und es liegt in der Verantwortung des Kunden, die mit den NFS v4.2-Sicherheitslabels und xattrs verbundenen Metadaten zu ermitteln. Details zur Aktualisierung und Aufbewahrung von Etiketten werden aus einfachen Grund weggelassen.

Beispiel PKI-Zertifikatwerte

Taste	Wert
EntitySecurityMark	t:s01 = NICHT KLASSIFIZIERT

Info	{
	"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"
	}
	}

Taste	Wert
Spezifikation	„DoD“
uuid	B4111349-7875-4115-ad30-0928565f2e15

AdminOrganisation	{
	"value": "DoD"
	}

Briefings	[
	{
	"value": "ABC1000"
	},
	{
	"value": "DEF1001"
	},
	{
	"value": "EFG2000"
	}
	]

Bürgerstatus	{
	"value": "US"
	}

Taste	Wert

Abstände	[
	{
	"value": "TS"
	},
	{
	"value": "S"
	},
	{
	"value": "C"
	},
	{
	"value": "U"
	}
	]

LänderOfMitgliedschaften	[
	{
	"value": "USA"
	}
	]

DigitalIdentifier	{
	"classification": "UNCLASSIFIED",
	"value": "cn=smith john r jrsmith, ou=dod, o=u.s. government, c=us"
	}

DissTos	{
	"value": "DoD"
	}

DytOrganisation	{
	"value": "DoD"
	}

Taste	Wert

EntityType	<pre>{
	"value": "GOV"
	}</pre>

FineAccessControls	<pre>[
	{
	"value": "SI"
	},
	{
	"value": "TK"
	},
	{
	"value": "NSYS"
	}
	]</pre>

Diese PKI-Berechtigungen zeigen die Zugangsdaten von John R. Smith, einschließlich des Zugriffs nach Datentyp und Zuordnung.

In Szenarien, in denen IC-TDF-Metadaten getrennt von der Datei gespeichert werden, empfiehlt NetApp eine zusätzliche Ebene feingranularer Zugriffskontrolle. Dabei werden Informationen zur Zugriffssteuerung sowohl auf Verzeichnisebene als auch in Verbindung mit jeder Datei gespeichert. Betrachten Sie als Beispiel die folgenden Tags, die mit einer Datei verknüpft sind:

- Sicherheitslabels für NFS v4.2: Werden für Sicherheitsentscheidungen verwendet
- Xattrs: Geben Sie ergänzende Informationen, die für die Datei und die Anforderungen an das organisatorische Programm relevant sind

Die folgenden Schlüssel-Wert-Paare sind Beispiele für Metadaten, die als xattrs gespeichert werden können und detaillierte Informationen über den Ersteller der Datei und die zugehörigen Sicherheitsklassifizierungen bieten. Diese Metadaten können von den Client-Applikationen genutzt werden, um fundierte Zugriffsentscheidungen zu treffen und Dateien gemäß den Standards und Anforderungen des Unternehmens zu organisieren.

Beispiel für xattr Schlüssel-Wert-Paare

Taste	Wert
`user.uuid`	`"761d2e3c-e778-4ee4-997b-3bb9a6a1d3fa"`
`user.entitySecurityMark`	`"UNCLASSIFIED"`

Taste	Wert
`user.specification`	`"INFO"`

Taste	Wert

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"
	},

Taste	Wert
`user.geo_point`	`[-78.7941, 35.7956]`

```
    }
}
```

Verwandte Informationen

- "NFS in NetApp ONTAP: Best Practice und Implementierungsleitfaden"
- "ONTAP-Befehlsreferenz"
- Anforderung von Kommentaren (RFC)
  - "RFC 7204: Anforderungen für gekennzeichnetes NFS"
  - "RFC 2203: RPCSEC_GSS-Protokollspezifikation"
  - "RFC 3530: Network File System (NFS) Version 4 Protocol"