Bucket- und Gruppenzugriffsrichtlinien

StorageGRID software

NetApp
October 21, 2025

Inhalt

Bucket- und Gruppenzugriffsrichtlinien

Verwenden Sie Bucket- und Gruppenzugriffsrichtlinien

StorageGRID verwendet die Richtliniensprache von Amazon Web Services (AWS), um S3-Mietern die Kontrolle über den Zugriff auf Buckets und Objekte in diesen Buckets zu ermöglichen. Das StorageGRID -System implementiert eine Teilmenge der S3 REST API-Richtliniensprache. Zugriffsrichtlinien für die S3-API sind in JSON geschrieben.

Übersicht über die Zugriffsrichtlinie

StorageGRID unterstützt zwei Arten von Zugriffsrichtlinien.

- **Bucket-Richtlinien**, die mithilfe der S3-API-Operationen GetBucketPolicy, PutBucketPolicy und DeleteBucketPolicy oder der Tenant Manager- oder Tenant Management-API verwaltet werden. Bucket-Richtlinien sind an Buckets angehängt und daher so konfiguriert, dass sie den Zugriff von Benutzern im Bucket-Eigentümerkonto oder anderen Konten auf den Bucket und die darin enthaltenen Objekte steuern. Eine Bucket-Richtlinie gilt nur für einen Bucket und möglicherweise für mehrere Gruppen.
- **Gruppenrichtlinien**, die mithilfe des Tenant Managers oder der Tenant Management API konfiguriert werden. Gruppenrichtlinien sind einer Gruppe im Konto zugeordnet und daher so konfiguriert, dass diese Gruppe auf bestimmte Ressourcen zugreifen kann, die diesem Konto gehören. Eine Gruppenrichtlinie gilt nur für eine Gruppe und möglicherweise mehrere Buckets.

Es gibt keinen Unterschied in der Priorität zwischen Gruppen- und Bucket-Richtlinien.

StorageGRID Bucket- und Gruppenrichtlinien folgen einer bestimmten, von Amazon definierten Grammatik. Innerhalb jeder Richtlinie befindet sich ein Array von Richtlinienanweisungen und jede Anweisung enthält die folgenden Elemente:

- Anweisungs-ID (Sid) (optional)
- Wirkung
- Auftraggeber/NichtAuftraggeber
- Ressource/NichtRessource
- Aktion/NichtAktion
- Bedingung (optional)

Richtlinienanweisungen werden mithilfe dieser Struktur erstellt, um Berechtigungen anzugeben: Gewähren Sie <Effekt>, um <Principal> die Ausführung von <Aktion> auf <Ressource> zu erlauben/verweigern, wenn <Bedingung> zutrifft.

Jedes Richtlinienelement wird für eine bestimmte Funktion verwendet:

Element	Beschreibung
Sid	Das Sid-Element ist optional. Die Sid dient lediglich als Beschreibung für den Benutzer. Es wird gespeichert, aber nicht vom StorageGRID -System interpretiert.

Element	Beschreibung
Wirkung	Verwenden Sie das Effect-Element, um festzulegen, ob die angegebenen Vorgänge zulässig oder verweigert werden. Sie müssen Vorgänge, die Sie für Buckets oder Objekte zulassen (oder verweigern), mithilfe der unterstützten Schlüsselwörter des Aktionselements identifizieren.
Auftraggeber/NichtAuftraggeber	Sie können Benutzern, Gruppen und Konten den Zugriff auf bestimmte Ressourcen und die Ausführung bestimmter Aktionen gestatten. Wenn in der Anfrage keine S3-Signatur enthalten ist, wird der anonyme Zugriff durch Angabe des Platzhalterzeichens (*) als Prinzipal zugelassen. Standardmäßig hat nur der Konto-Root Zugriff auf die Ressourcen, die dem Konto gehören. Sie müssen nur das Principal-Element in einer Bucket-Richtlinie angeben. Bei Gruppenrichtlinien ist die Gruppe, an die die Richtlinie angehängt ist, das implizite Principal-Element.
Ressource/NichtRessource	Das Ressourcenelement identifiziert Buckets und Objekte. Sie können Berechtigungen für Buckets und Objekte erteilen oder verweigern, indem Sie den Amazon Resource Name (ARN) zur Identifizierung der Ressource verwenden.
Aktion/NichtAktion	Die Elemente „Aktion“ und „Effekt“ sind die beiden Komponenten von Berechtigungen. Wenn eine Gruppe eine Ressource anfordert, wird ihr der Zugriff auf die Ressource entweder gewährt oder verweigert. Der Zugriff wird verweigert, sofern Sie keine ausdrücklichen Berechtigungen erteilen. Sie können jedoch eine durch eine andere Richtlinie erteilte Berechtigung durch eine explizite Verweigerung außer Kraft setzen.
Zustand	Das Bedingungselement ist optional. Mithilfe von Bedingungen können Sie Ausdrücke erstellen, um zu bestimmen, wann eine Richtlinie angewendet werden soll.

Im Aktionselement können Sie das Platzhalterzeichen (*) verwenden, um alle Vorgänge oder eine Teilmenge von Vorgängen anzugeben. Diese Aktion entspricht beispielsweise Berechtigungen wie s3:GetObject, s3:PutObject und s3:DeleteObject.

```
s3:*Object
```

Im Ressourcenelement können Sie die Platzhalterzeichen (*) und (?) verwenden. Während das Sternchen (*) 0 oder mehr Zeichen entspricht, entspricht das Fragezeichen (?) einem beliebigen einzelnen Zeichen.

Im Principal-Element werden Platzhalterzeichen nur zum Festlegen des anonymen Zugriffs unterstützt, der jedem die Berechtigung erteilt. Beispielsweise legen Sie das Platzhalterzeichen (*) als Hauptwert fest.

```
"Principal":"*"
```

```
"Principal":{"AWS":"*"}
```

Im folgenden Beispiel verwendet die Anweisung die Elemente „Effect“, „Principal“, „Action“ und „Resource“. Dieses Beispiel zeigt eine vollständige Bucket-Richtlinienanweisung, die den Effekt "Zulassen" verwendet, um den Principals, der Admin-Gruppe `federated-group/admin` und die Finanzgruppe `federated-group/finance` , Berechtigungen zum Ausführen der Aktion `s3:ListBucket` auf dem Eimer namens `mybucket` und die Aktion `s3:GetObject` auf allen Objekten in diesem Bucket.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::27233906934684427525:federated-group/admin",
          "arn:aws:iam::27233906934684427525:federated-group/finance"
        ]
      },
      "Action": [
        "s3:ListBucket",
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::mybucket",
        "arn:aws:s3:::mybucket/*"
      ]
    }
  ]
}
```

Die Bucket-Richtlinie hat eine Größenbeschränkung von 20.480 Bytes und die Gruppenrichtlinie eine Größenbeschränkung von 5.120 Bytes.

Konsistenz für Richtlinien

Standardmäßig sind alle Aktualisierungen, die Sie an Gruppenrichtlinien vornehmen, letztendlich konsistent. Wenn eine Gruppenrichtlinie konsistent wird, kann es aufgrund der Richtlinienzwischenspeicherung weitere 15 Minuten dauern, bis die Änderungen wirksam werden. Standardmäßig sind alle Aktualisierungen, die Sie an Bucket-Richtlinien vornehmen, streng konsistent.

Bei Bedarf können Sie die Konsistenzgarantien für Bucket-Richtlinienaktualisierungen ändern. Beispielsweise möchten Sie möglicherweise, dass eine Änderung an einer Bucket-Richtlinie während eines Site-Ausfalls verfügbar ist.

In diesem Fall können Sie entweder die `Consistency-Control` Header in der PutBucketPolicy-Anforderung, oder Sie können die PUT Bucket-Konsistenzanforderung verwenden. Wenn eine Bucket-Richtlinie konsistent wird, kann es aufgrund der Richtlinienzwischenspeicherung weitere 8 Sekunden dauern,

bis die Änderungen wirksam werden.

> Wenn Sie die Konsistenz auf einen anderen Wert einstellen, um eine vorübergehende Situation zu umgehen, denken Sie daran, die Einstellung auf Bucket-Ebene wieder auf den ursprünglichen Wert zurückzusetzen, wenn Sie fertig sind. Andernfalls verwenden alle zukünftigen Bucket-Anfragen die geänderte Einstellung.

Verwenden Sie ARN in Richtlinienanweisungen

In Richtlinienanweisungen wird die ARN in den Elementen „Principal“ und „Resource“ verwendet.

- Verwenden Sie diese Syntax, um die S3-Ressourcen-ARN anzugeben:

```
arn:aws:s3:::bucket-name
arn:aws:s3:::bucket-name/object_key
```

- Verwenden Sie diese Syntax, um die ARN der Identitätsressource (Benutzer und Gruppen) anzugeben:

```
arn:aws:iam::account_id:root
arn:aws:iam::account_id:user/user_name
arn:aws:iam::account_id:group/group_name
arn:aws:iam::account_id:federated-user/user_name
arn:aws:iam::account_id:federated-group/group_name
```

Weitere Überlegungen:

- Sie können das Sternchen (*) als Platzhalter verwenden, um null oder mehr Zeichen im Objektschlüssel abzugleichen.
- Internationale Zeichen, die im Objektschlüssel angegeben werden können, sollten mit JSON UTF-8 oder mit JSON \u-Escapesequenzen codiert werden. Prozentkodierung wird nicht unterstützt.

  "RFC 2141 URN-Syntax"

  Der HTTP-Anforderungstext für den PutBucketPolicy-Vorgang muss mit charset=UTF-8 codiert sein.

Angeben von Ressourcen in einer Richtlinie

In Richtlinienanweisungen können Sie das Ressourcenelement verwenden, um den Bucket oder das Objekt anzugeben, für das Berechtigungen erteilt oder verweigert werden.

- Jede Richtlinienanweisung erfordert ein Ressourcenelement. In einer Richtlinie werden Ressourcen durch das Element gekennzeichnet `Resource` oder alternativ `NotResource` zum Ausschluss.
- Sie geben Ressourcen mit einer S3-Ressourcen-ARN an. Beispiel:

```
"Resource": "arn:aws:s3:::mybucket/*"
```

- Sie können auch Richtlinienvariablen innerhalb des Objektschlüssels verwenden. Beispiel:

```
"Resource": "arn:aws:s3:::mybucket/home/${aws:username}/*"
```

- Der Ressourcenwert kann einen Bucket angeben, der beim Erstellen einer Gruppenrichtlinie noch nicht vorhanden ist.

Angeben von Prinzipalen in einer Richtlinie

Verwenden Sie das Principal-Element, um den Benutzer, die Gruppe oder das Mandantenkonto zu identifizieren, dem durch die Richtlinienanweisung der Zugriff auf die Ressource gestattet bzw. verweigert wird.

- Jede Richtlinienanweisung in einer Bucket-Richtlinie muss ein Principal-Element enthalten. Richtlinienanweisungen in einer Gruppenrichtlinie benötigen das Principal-Element nicht, da die Gruppe als Auftraggeber verstanden wird.
- In einer Richtlinie werden Auftraggeber durch das Element „Principal“ oder alternativ „NotPrincipal“ zum Ausschluss gekennzeichnet.
- Kontobasierte Identitäten müssen mithilfe einer ID oder einer ARN angegeben werden:

```
"Principal": { "AWS": "account_id"}
"Principal": { "AWS": "identity_arn" }
```

- In diesem Beispiel wird die Mandantenkonto-ID 27233906934684427525 verwendet, die das Stammkonto und alle Benutzer im Konto umfasst:

```
 "Principal": { "AWS": "27233906934684427525" }
```

- Sie können nur das Stammkonto angeben:

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:root" }
```

- Sie können einen bestimmten Verbundbenutzer („Alex“) angeben:

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:federated-
user/Alex" }
```

- Sie können eine bestimmte föderierte Gruppe („Manager“) angeben:

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:federated-
group/Managers"  }
```

- Sie können einen anonymen Auftraggeber angeben:

```
"Principal": "*"
```

- Um Mehrdeutigkeiten zu vermeiden, können Sie anstelle des Benutzernamens die Benutzer-UUID verwenden:

```
arn:aws:iam::27233906934684427525:user-uuid/de305d54-75b4-431b-adb2-eb6b9e546013
```

  Nehmen wir beispielsweise an, Alex verlässt die Organisation und der Benutzername `Alex` wird gelöscht. Wenn ein neuer Alex in die Organisation eintritt und ihm die gleiche `Alex` Benutzernamen, könnte der neue Benutzer unbeabsichtigt die dem ursprünglichen Benutzer erteilten Berechtigungen erben.

- Der Prinzipalwert kann einen Gruppen-/Benutzernamen angeben, der beim Erstellen einer Bucket-Richtlinie noch nicht vorhanden ist.

Festlegen von Berechtigungen in einer Richtlinie

In einer Richtlinie wird das Aktionselement verwendet, um Berechtigungen für eine Ressource zuzulassen/zu verweigern. Es gibt eine Reihe von Berechtigungen, die Sie in einer Richtlinie angeben können. Diese werden durch das Element „Action“ oder alternativ „NotAction“ zum Ausschluss gekennzeichnet. Jedes dieser Elemente ist bestimmten S3 REST-API-Operationen zugeordnet.

In den Tabellen sind die Berechtigungen aufgeführt, die für Buckets gelten, und die Berechtigungen, die für Objekte gelten.

Amazon S3 verwendet jetzt die Berechtigung s3:PutReplicationConfiguration sowohl für die Aktionen PutBucketReplication als auch DeleteBucketReplication. StorageGRID verwendet für jede Aktion separate Berechtigungen, was der ursprünglichen Amazon S3-Spezifikation entspricht.

Ein Löschen wird ausgeführt, wenn ein Put zum Überschreiben eines vorhandenen Werts verwendet wird.

Berechtigungen, die für Buckets gelten

Berechtigungen	S3 REST API-Operationen	Benutzerdefiniert für StorageGRID
s3:Bucket erstellen	Bucket erstellen	Ja. **Hinweis**: Nur in Gruppenrichtlinien verwenden.
s3:Bucket löschen	Bucket löschen	
s3:DeleteBucketMetadataNotification	Konfiguration der Benachrichtigung über DELETE-Bucket-Metadaten	Ja

Berechtigungen	S3 REST API-Operationen	Benutzerdefiniert für StorageGRID
s3:DeleteBucketPolicy	DeleteBucketPolicy	
s3:Replikationskonfiguration löschen	DeleteBucketReplication	Ja, separate Berechtigungen für PUT und DELETE
s3:GetBucketAcl	GetBucketAcl	
s3:GetBucketCompliance	GET Bucket-Konformität (veraltet)	Ja
s3:GetBucketConsistency	GET Bucket-Konsistenz	Ja
s3:GetBucketCORS	GetBucketCors	
s3:GetEncryptionConfiguration	GetBucketEncryption	
s3:GetBucketLastAccessTime	GET Bucket – Letzte Zugriffszeit	Ja
s3:GetBucketLocation	BucketLocation abrufen	
s3:GetBucketMetadataNotification	GET Bucket-Metadaten-Benachrichtigungskonfiguration	Ja
s3:GetBucketNotification	GetBucketNotificationConfiguration	
s3:GetBucketObjectLockConfiguration	GetObjectLockConfiguration	
s3:GetBucketPolicy	GetBucketPolicy	
s3:GetBucketTagging	GetBucketTagging	
s3:GetBucketVersioning	GetBucketVersioning	
s3:GetLifecycleConfiguration	GetBucketLifecycleConfiguration	
s3:GetReplicationConfiguration	GetBucketReplication	
s3:ListeAlleMeineBuckets	• Buckets auflisten • GET-Speichernutzung	Ja, für die GET-Speichernutzung. **Hinweis**: Nur in Gruppenrichtlinien verwenden.

Berechtigungen	S3 REST API-Operationen	Benutzerdefiniert für StorageGRID
s3:ListBucket	• ListObjects • Kopfeimer • RestoreObject	
s3:ListBucketMultipartUploads	• ListMultipartUploads • RestoreObject	
s3:ListBucketVersions	GET Bucket-Versionen	
s3:PutBucketCompliance	PUT-Bucket-Konformität (veraltet)	Ja
s3:PutBucketConsistency	PUT Bucket-Konsistenz	Ja
s3:PutBucketCORS	• DeleteBucketCors† • PutBucketCors	
s3:PutEncryptionConfiguration	• DeleteBucketEncryption • PutBucketEncryption	
s3:PutBucketLastAccessTime	PUT Bucket: Letzte Zugriffszeit	Ja
s3:PutBucketMetadataNotification	Konfiguration der Benachrichtigung über PUT-Bucket-Metadaten	Ja
s3:PutBucketNotification	PutBucketNotificationConfiguration	
s3:PutBucketObjectLockConfiguration	• CreateBucket mit dem `x-amz-bucket-object-lock-enabled: true` Anforderungsheader (erfordert auch die Berechtigung s3:CreateBucket) • PutObjectLockConfiguration	
s3:PutBucketPolicy	PutBucketPolicy	
s3:PutBucketTagging	• BucketTagging löschen† • PutBucketTagging	
s3:PutBucketVersioning	PutBucketVersioning	

Berechtigungen	S3 REST API-Operationen	Benutzerdefiniert für StorageGRID
s3:PutLifecycleConfiguration	• DeleteBucketLifecycle† • PutBucketLifecycleConfiguration	
s3:PutReplicationConfiguration	PutBucketReplication	Ja, separate Berechtigungen für PUT und DELETE

Berechtigungen, die für Objekte gelten

Berechtigungen	S3 REST API-Operationen	Benutzerdefiniert für StorageGRID
s3:AbortMultipartUpload	• AbortMultipartUpload • RestoreObject	
s3:BypassGovernanceRetention	• Objekt löschen • Objekte löschen • PutObjectRetention	
s3:Objekt löschen	• Objekt löschen • Objekte löschen • RestoreObject	
s3:DeleteObjectTagging	DeleteObjectTagging	
s3:DeleteObjectVersionTagging	DeleteObjectTagging (eine bestimmte Version des Objekts)	
s3:DeleteObjectVersion	DeleteObject (eine bestimmte Version des Objekts)	
s3:GetObject	• GetObject • HeadObject • RestoreObject • SelectObjectContent	
s3:GetObjectAcl	GetObjectAcl	
s3:GetObjectLegalHold	GetObjectLegalHold	
s3:GetObjectRetention	GetObjectRetention	

Berechtigungen	S3 REST API-Operationen	Benutzerdefiniert für StorageGRID
s3:GetObjectTagging	GetObjectTagging	
s3:GetObjectVersionTagging	GetObjectTagging (eine bestimmte Version des Objekts)	
s3:GetObjectVersion	GetObject (eine bestimmte Version des Objekts)	
s3:ListMultipartUploadParts	ListParts, RestoreObject	
s3:PutObject	• PutObject • Objekt kopieren • RestoreObject • CreateMultipartUpload • CompleteMultipartUpload • UploadPart • UploadPartCopy	
s3:PutObjectLegalHold	PutObjectLegalHold	
s3:PutObjectRetention	PutObjectRetention	
s3:PutObjectTagging	PutObjectTagging	
s3:PutObjectVersionTagging	PutObjectTagging (eine bestimmte Version des Objekts)	
s3:PutOverwriteObject	• PutObject • Objekt kopieren • PutObjectTagging • DeleteObjectTagging • CompleteMultipartUpload	Ja
s3:RestoreObject	RestoreObject	

PutOverwriteObject-Berechtigung verwenden

Die Berechtigung s3:PutOverwriteObject ist eine benutzerdefinierte StorageGRID Berechtigung, die für Vorgänge gilt, die Objekte erstellen oder aktualisieren. Die Einstellung dieser Berechtigung bestimmt, ob der Client die Daten, benutzerdefinierten Metadaten oder S3-Objektmarkierungen eines Objekts überschreiben kann.

Mögliche Einstellungen für diese Berechtigung sind:

- **Zulassen**: Der Client kann ein Objekt überschreiben. Dies ist die Standardeinstellung.
- **Ablehnen**: Der Client kann ein Objekt nicht überschreiben. Wenn die Berechtigung „PutOverwriteObject“ auf „Verweigern“ gesetzt ist, funktioniert sie wie folgt:
  - Wenn ein vorhandenes Objekt am gleichen Pfad gefunden wird:
    - Die Daten, benutzerdefinierten Metadaten oder S3-Objektmarkierungen des Objekts können nicht überschrieben werden.
    - Alle laufenden Aufnahmevorgänge werden abgebrochen und ein Fehler zurückgegeben.
    - Wenn die S3-Versionierung aktiviert ist, verhindert die Einstellung „Verweigern“, dass PutObjectTagging- oder DeleteObjectTagging-Vorgänge das TagSet für ein Objekt und seine nicht aktuellen Versionen ändern.
  - Wenn ein vorhandenes Objekt nicht gefunden wird, hat diese Berechtigung keine Wirkung.
- Wenn diese Berechtigung nicht vorhanden ist, ist die Wirkung dieselbe, als ob „Zulassen“ gesetzt wäre.

> Wenn die aktuelle S3-Richtlinie das Überschreiben zulässt und die Berechtigung „PutOverwriteObject“ auf „Verweigern“ gesetzt ist, kann der Client die Daten, benutzerdefinierten Metadaten oder Objektmarkierungen eines Objekts nicht überschreiben. Wenn außerdem das Kontrollkästchen **Client-Änderung verhindern** aktiviert ist (**KONFIGURATION** > **Sicherheitseinstellungen** > **Netzwerk und Objekte**), überschreibt diese Einstellung die Einstellung der Berechtigung „PutOverwriteObject“.

Bedingungen in einer Richtlinie angeben

Bedingungen definieren, wann eine Richtlinie in Kraft tritt. Bedingungen bestehen aus Operatoren und Schlüssel-Wert-Paaren.

Bedingungen verwenden Schlüssel-Wert-Paare zur Auswertung. Ein Bedingungselement kann mehrere Bedingungen enthalten und jede Bedingung kann mehrere Schlüssel-Wert-Paare enthalten. Der Bedingungsblock verwendet das folgende Format:

```
Condition: {
     condition_type: {
          condition_key: condition_values
```

Im folgenden Beispiel verwendet die Bedingung „IpAddress“ den Bedingungsschlüssel „SourceIp“.

```
"Condition": {
    "IpAddress": {
      "aws:SourceIp": "54.240.143.0/24"
        ...
},
        ...
```

Unterstützte Bedingungsoperatoren

Bedingungsoperatoren werden wie folgt kategorisiert:

- Zeichenfolge
- Numerisch
- Boolescher Wert
- IP-Adresse
- Nullprüfung

Bedingungsoperatoren	Beschreibung
StringEquals	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert auf Basis einer genauen Übereinstimmung (Groß-/Kleinschreibung beachten).
StringNotEquals	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert basierend auf negierter Übereinstimmung (Groß-/Kleinschreibung beachten).
StringEqualsIgnoreCase	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert basierend auf einer genauen Übereinstimmung (Groß-/Kleinschreibung wird ignoriert).
StringNotEqualsIgnoreCase	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert basierend auf negierter Übereinstimmung (Groß-/Kleinschreibung wird ignoriert).
StringLike	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert auf Basis einer genauen Übereinstimmung (Groß-/Kleinschreibung beachten). Kann die Platzhalterzeichen * und ? enthalten.
StringNotLike	Vergleicht einen Schlüssel mit einem Zeichenfolgenwert basierend auf negierter Übereinstimmung (Groß-/Kleinschreibung beachten). Kann die Platzhalterzeichen * und ? enthalten.
NumericEquals	Vergleicht einen Schlüssel mit einem numerischen Wert auf Basis einer exakten Übereinstimmung.
NumericNotEquals	Vergleicht einen Schlüssel mit einem numerischen Wert basierend auf negierter Übereinstimmung.
NumerischGrößerAls	Vergleicht einen Schlüssel mit einem numerischen Wert basierend auf einer „Größer-als“-Übereinstimmung.
NumerischGrößerAlsGleich	Vergleicht einen Schlüssel mit einem numerischen Wert basierend auf der Übereinstimmung „größer als oder gleich“.
NumericLessThan	Vergleicht einen Schlüssel mit einem numerischen Wert basierend auf einer „kleiner als“-Übereinstimmung.

Bedingungsoperatoren	Beschreibung
NumerischKleinerAlsGleich	Vergleicht einen Schlüssel mit einem numerischen Wert basierend auf der Übereinstimmung „kleiner als oder gleich“.
Bool	Vergleicht einen Schlüssel mit einem Booleschen Wert basierend auf der Übereinstimmung „wahr oder falsch“.
IP-Adresse	Vergleicht einen Schlüssel mit einer IP-Adresse oder einem IP-Adressbereich.
NotIpAddress	Vergleicht einen Schlüssel mit einer IP-Adresse oder einem IP-Adressbereich basierend auf negierter Übereinstimmung.
Null	Überprüft, ob im aktuellen Anforderungskontext ein Bedingungsschlüssel vorhanden ist.

Unterstützte Bedingungsschlüssel

Bedingungsschlüssel	Aktionen	Beschreibung
aws:SourceIp	IP-Betreiber	Wird mit der IP-Adresse verglichen, von der die Anfrage gesendet wurde. Kann für Bucket- oder Objektoperationen verwendet werden. **Hinweis:** Wenn die S3-Anforderung über den Load Balancer-Dienst auf Admin-Knoten und Gateway-Knoten gesendet wurde, wird dies mit der IP-Adresse vor dem Load Balancer-Dienst verglichen. **Hinweis**: Wenn ein nicht transparenter Load Balancer eines Drittanbieters verwendet wird, wird dies mit der IP-Adresse dieses Load Balancers verglichen. Beliebig `X-Forwarded-For` Header wird ignoriert, da seine Gültigkeit nicht festgestellt werden kann.
aws:Benutzername	Ressource/Identität	Wird mit dem Benutzernamen des Absenders verglichen, von dem die Anfrage gesendet wurde. Kann für Bucket- oder Objektoperationen verwendet werden.
s3:Trennzeichen	s3:ListBucket und s3:ListBucketVersions-Berechtigungen	Wird mit dem in einer ListObjects- oder ListObjectVersions-Anforderung angegebenen Trennzeichenparameter verglichen.

Bedingungsschlüssel	Aktionen	Beschreibung
s3:ExistingObjectTag/<Tag-Schlüssel>	s3:DeleteObjectTagging s3:DeleteObjectVersionTagging s3:GetObject s3:GetObjectAcl 3:GetObjectTagging s3:GetObjectVersion s3:GetObjectVersionAcl s3:GetObjectVersionTagging s3:PutObjectAcl s3:PutObjectTagging s3:PutObjectVersionAcl s3:PutObjectVersionTagging	Erfordert, dass das vorhandene Objekt über den spezifischen Tag-Schlüssel und -Wert verfügt.
s3:max-Schlüssel	s3:ListBucket und s3:ListBucketVersions-Berechtigungen	Wird mit dem in einer ListObjects- oder ListObjectVersions-Anforderung angegebenen Max-Keys-Parameter verglichen.
s3:Objektsperre-verbleibende-Aufbewahrungstage	s3:PutObject	Vergleicht mit dem Aufbewahrungsdatum, das in der `x-amz-object-lock-retain-until-date` Anforderungsheader oder berechnet aus der Standardaufbewahrungsdauer des Buckets, um sicherzustellen, dass diese Werte innerhalb des zulässigen Bereichs für die folgenden Anforderungen liegen: • PutObject • Objekt kopieren • CreateMultipartUpload
s3:Objektsperre-verbleibende-Aufbewahrungstage	s3:PutObjectRetention	Vergleicht mit dem in der PutObjectRetention-Anforderung angegebenen Aufbewahrungsdatum, um sicherzustellen, dass es innerhalb des zulässigen Bereichs liegt.

Bedingungsschlüssel	Aktionen	Beschreibung
s3:Präfix	s3:ListBucket und s3:ListBucketVersions-Berechtigungen	Wird mit dem in einer ListObjects- oder ListObjectVersions-Anforderung angegebenen Präfixparameter verglichen.
s3:RequestObjectTag/<Tag-Schlüssel>	s3:PutObject s3:PutObjectTagging s3:PutObjectVersionTagging	Erfordert einen bestimmten Tag-Schlüssel und -Wert, wenn die Objektanforderung Tagging enthält.

Angeben von Variablen in einer Richtlinie

Sie können Variablen in Richtlinien verwenden, um Richtlinieninformationen einzufügen, wenn diese verfügbar sind. Sie können Richtlinienvariablen in der `Resource` Element und in Stringvergleichen im `Condition` Element.

In diesem Beispiel ist die Variable `${aws:username}` ist Teil des Ressourcenelements:

```
"Resource": "arn:aws:s3:::bucket-name/home/${aws:username}/*"
```

In diesem Beispiel ist die Variable `${aws:username}` ist Teil des Bedingungswerts im Bedingungsblock:

```
"Condition": {
    "StringLike": {
      "s3:prefix": "${aws:username}/*"
        ...
},
        ...
```

Variable	Beschreibung
`${aws:SourceIp}`	Verwendet den SourceIp-Schlüssel als bereitgestellte Variable.
`${aws:username}`	Verwendet den Benutzernamenschlüssel als bereitgestellte Variable.
`${s3:prefix}`	Verwendet den dienstspezifischen Präfixschlüssel als bereitgestellte Variable.
`${s3:max-keys}`	Verwendet den dienstspezifischen Max-Keys-Schlüssel als bereitgestellte Variable.

Variable	Beschreibung
`${*}`	Sonderzeichen. Verwendet das Zeichen als wörtliches *-Zeichen.
`${?}`	Sonderzeichen. Verwendet das Zeichen als wörtliches ?-Zeichen.
`${$}`	Sonderzeichen. Verwendet das Zeichen als wörtliches $-Zeichen.

Erstellen Sie Richtlinien, die eine besondere Behandlung erfordern

Manchmal kann eine Richtlinie Berechtigungen erteilen, die eine Gefahr für die Sicherheit oder den laufenden Betrieb darstellen, wie etwa das Sperren des Root-Benutzers des Kontos. Die StorageGRID S3 REST-API-Implementierung ist bei der Richtlinienvalidierung weniger restriktiv als Amazon, bei der Richtlinienauswertung jedoch ebenso streng.

Richtlinienbeschreibung	Richtlinientyp	Amazon-Verhalten	StorageGRID -Verhalten
Verweigern Sie sich selbst alle Berechtigungen für das Root-Konto	Eimer	Gültig und erzwungen, aber das Root-Benutzerkonto behält die Berechtigung für alle S3-Bucket-Richtlinienvorgänge	Dasselbe
Sich selbst alle Berechtigungen für Benutzer/Gruppe verweigern	Gruppe	Gültig und durchgesetzt	Dasselbe
Erteilen Sie einer fremden Kontogruppe alle Berechtigungen	Eimer	Ungültiger Auftraggeber	Gültig, aber Berechtigungen für alle S3-Bucket-Richtlinienvorgänge geben einen 405-Methodenfehler zurück, wenn sie durch eine Richtlinie erlaubt sind
Erteilen Sie einem fremden Root- oder Benutzerkonto alle Berechtigungen	Eimer	Gültig, aber Berechtigungen für alle S3-Bucket-Richtlinienvorgänge geben einen 405-Methodenfehler zurück, wenn sie durch eine Richtlinie erlaubt sind	Dasselbe
Jedem die Berechtigung für alle Aktionen erteilen	Eimer	Gültig, aber Berechtigungen für alle S3-Bucket-Richtlinienvorgänge geben einen 405-Methode nicht zulässig-Fehler für das Stammkonto und die Benutzer des Fremdkontos zurück	Dasselbe

Richtlinienbeschreibung	Richtlinientyp	Amazon-Verhalten	StorageGRID -Verhalten
Allen die Berechtigung für alle Aktionen verweigern	Eimer	Gültig und erzwungen, aber das Root-Benutzerkonto behält die Berechtigung für alle S3-Bucket-Richtlinienvorgänge	Dasselbe
Der Auftraggeber ist ein nicht vorhandener Benutzer oder eine nicht vorhandene Gruppe.	Eimer	Ungültiger Auftraggeber	Gültig
Ressource ist ein nicht vorhandener S3-Bucket	Gruppe	Gültig	Dasselbe
Principal ist eine lokale Gruppe	Eimer	Ungültiger Auftraggeber	Gültig
Die Richtlinie erteilt Nichtbesitzerkonten (einschließlich anonymer Konten) die Berechtigung, Objekte abzulegen.	Eimer	Gültig. Objekte sind Eigentum des Erstellerkontos und die Bucket-Richtlinie gilt nicht. Das Erstellerkonto muss mithilfe von Objekt-ACLs Zugriffsberechtigungen für das Objekt erteilen.	Gültig. Objekte sind Eigentum des Bucket-Eigentümerkontos. Es gilt die Bucket-Richtlinie.

WORM-Schutz (Write-Once-Read-Many)

Sie können WORM-Buckets (Write-Once-Read-Many) erstellen, um Daten, benutzerdefinierte Objektmetadaten und S3-Objekt-Tagging zu schützen. Sie konfigurieren die WORM-Buckets, um die Erstellung neuer Objekte zu ermöglichen und das Überschreiben oder Löschen vorhandener Inhalte zu verhindern. Verwenden Sie einen der hier beschriebenen Ansätze.

Um sicherzustellen, dass Überschreibungen immer verweigert werden, können Sie:

- Gehen Sie im Grid Manager zu **KONFIGURATION** > **Sicherheit** > **Sicherheitseinstellungen** > **Netzwerk und Objekte** und aktivieren Sie das Kontrollkästchen **Client-Änderung verhindern**.
- Wenden Sie die folgenden Regeln und S3-Richtlinien an:
  - Fügen Sie der S3-Richtlinie eine PutOverwriteObject DENY-Operation hinzu.
  - Fügen Sie der S3-Richtlinie eine DeleteObject DENY-Operation hinzu.
  - Fügen Sie der S3-Richtlinie eine PutObject ALLOW-Operation hinzu.

> Das Festlegen von „DeleteObject“ auf „DENY“ in einer S3-Richtlinie verhindert nicht, dass ILM Objekte löscht, wenn eine Regel wie „Null Kopien nach 30 Tagen“ vorhanden ist.

Selbst wenn alle diese Regeln und Richtlinien angewendet werden, schützen sie nicht vor gleichzeitigen Schreibvorgängen (siehe Situation A). Sie schützen vor sequenziellen Überschreibungen (siehe Situation B).

Situation A: Gleichzeitige Schreibvorgänge (nicht geschützt)

```
/mybucket/important.doc
PUT#1 ---> OK
PUT#2 -------> OK
```

Situation B: Sequentielles Überschreiben abgeschlossen (vorbeugend)

```
/mybucket/important.doc
PUT#1 -------> PUT#2 ---X (denied)
```

Ähnliche Informationen

- "So verwalten StorageGRID ILM-Regeln Objekte"
- "Beispiele für Bucket-Richtlinien"
- "Beispiele für Gruppenrichtlinien"
- "Objekte mit ILM verwalten"
- "Verwenden eines Mandantenkontos"

Beispiele für Bucket-Richtlinien

Verwenden Sie die Beispiele in diesem Abschnitt, um StorageGRID Zugriffsrichtlinien für Buckets zu erstellen.

Bucket-Richtlinien geben die Zugriffsberechtigungen für den Bucket an, an den die Richtlinie angehängt ist. Sie konfigurieren eine Bucket-Richtlinie mithilfe der S3 PutBucketPolicy-API über eines dieser Tools:

- "Mietermanager" .
- AWS CLI mit diesem Befehl (siehe"Operationen an Buckets"):

```
> aws s3api put-bucket-policy --bucket examplebucket --policy
file://policy.json
```

Beispiel: Allen Lesezugriff auf einen Bucket gewähren

In diesem Beispiel darf jeder, auch anonyme Benutzer, Objekte im Bucket auflisten und GetObject-Operationen für alle Objekte im Bucket ausführen. Alle anderen Vorgänge werden abgelehnt. Beachten Sie, dass diese Richtlinie möglicherweise nicht besonders nützlich ist, da niemand außer dem Konto-Root über die Berechtigung zum Schreiben in den Bucket verfügt.

```
{
  "Statement": [
    {
      "Sid": "AllowEveryoneReadOnlyAccess",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [ "s3:GetObject", "s3:ListBucket" ],
      "Resource":
["arn:aws:s3:::examplebucket","arn:aws:s3:::examplebucket/*"]
    }
  ]
}
```

Beispiel: Erlauben Sie allen Benutzern eines Kontos den Vollzugriff und allen Benutzern eines anderen Kontos den Lesezugriff auf einen Bucket.

In diesem Beispiel erhält jeder in einem angegebenen Konto vollen Zugriff auf einen Bucket, während jeder in einem anderen angegebenen Konto nur den Bucket auflisten und GetObject-Operationen für Objekte im Bucket ausführen darf, beginnend mit dem `shared/` Objektschlüsselpräfix.

> In StorageGRID sind Objekte, die von einem Nicht-Eigentümerkonto (einschließlich anonymer Konten) erstellt wurden, Eigentum des Bucket-Eigentümerkontos. Für diese Objekte gilt die Bucket-Richtlinie.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "95390887230002558202"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "31181711887329436680"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::examplebucket/shared/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "31181711887329436680"
      },
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::examplebucket",
      "Condition": {
        "StringLike": {
          "s3:prefix": "shared/*"
        }
      }
    }
  ]
}
```

Beispiel: Erlauben Sie allen nur Lesezugriff auf einen Bucket und Vollzugriff für eine bestimmte Gruppe

In diesem Beispiel darf jeder, auch anonym, den Bucket auflisten und GetObject-Operationen für alle Objekte im Bucket durchführen, während nur Benutzer der Gruppe `Marketing` im angegebenen Konto wird der volle Zugriff gewährt.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/Marketing"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": ["s3:ListBucket","s3:GetObject"],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    }
  ]
}
```

Beispiel: Erlauben Sie jedem Lese- und Schreibzugriff auf einen Bucket, wenn sich der Client im IP-Bereich befindet

In diesem Beispiel darf jeder, auch anonyme Benutzer, den Bucket auflisten und beliebige Objektoperationen für alle Objekte im Bucket ausführen, vorausgesetzt, die Anforderungen stammen aus einem angegebenen IP-Bereich (54.240.143.0 bis 54.240.143.255, außer 54.240.143.188). Alle anderen Vorgänge werden abgelehnt und alle Anfragen außerhalb des IP-Bereichs werden abgelehnt.

```
{
  "Statement": [
    {
      "Sid": "AllowEveryoneReadWriteAccessIfInSourceIpRange",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [ "s3:*Object", "s3:ListBucket" ],
      "Resource":
["arn:aws:s3:::examplebucket","arn:aws:s3:::examplebucket/*"],
      "Condition": {
        "IpAddress": {"aws:SourceIp": "54.240.143.0/24"},
        "NotIpAddress": {"aws:SourceIp": "54.240.143.188"}
      }
    }
  ]
}
```

Beispiel: Vollzugriff auf einen Bucket ausschließlich durch einen angegebenen Verbundbenutzer zulassen

In diesem Beispiel erhält der Verbundbenutzer Alex vollen Zugriff auf die `examplebucket` Bucket und seine Objekte. Allen anderen Benutzern, einschließlich „root“, werden sämtliche Vorgänge ausdrücklich verweigert. Beachten Sie jedoch, dass „root“ niemals die Berechtigung zum Put/Get/DeleteBucketPolicy verweigert wird.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-user/Alex"
      },
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Deny",
      "NotPrincipal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-user/Alex"
      },
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    }
  ]
}
```

Beispiel: PutOverwriteObject-Berechtigung

In diesem Beispiel `Deny` Die Wirkung von PutOverwriteObject und DeleteObject stellt sicher, dass niemand die Daten, benutzerdefinierten Metadaten und S3-Objektmarkierungen des Objekts überschreiben oder löschen kann.

```
{
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": [
        "s3:PutOverwriteObject",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion"
      ],
      "Resource": "arn:aws:s3:::wormbucket/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/SomeGroup"

},
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::wormbucket"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/SomeGroup"

},
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::wormbucket/*"
    }
  ]
}
```

Beispiele für Gruppenrichtlinien

Verwenden Sie die Beispiele in diesem Abschnitt, um StorageGRID Zugriffsrichtlinien für Gruppen zu erstellen.

Gruppenrichtlinien geben die Zugriffsberechtigungen für die Gruppe an, der die Richtlinie zugeordnet ist. Es gibt keine `Principal` Element in der Richtlinie, da es implizit ist. Gruppenrichtlinien werden mithilfe des Tenant Managers oder der API konfiguriert.

Beispiel: Festlegen der Gruppenrichtlinie mit dem Mandanten-Manager

Wenn Sie im Mandanten-Manager eine Gruppe hinzufügen oder bearbeiten, können Sie eine Gruppenrichtlinie auswählen, um festzulegen, welche S3-Zugriffsberechtigungen die Mitglieder dieser Gruppe haben. Sehen "Erstellen von Gruppen für einen S3-Mandanten" .

- **Kein S3-Zugriff**: Standardoption. Benutzer in dieser Gruppe haben keinen Zugriff auf S3-Ressourcen, es sei denn, der Zugriff wird mit einer Bucket-Richtlinie gewährt. Wenn Sie diese Option auswählen, hat standardmäßig nur der Root-Benutzer Zugriff auf S3-Ressourcen.
- **Nur-Lesezugriff**: Benutzer in dieser Gruppe haben nur Lesezugriff auf S3-Ressourcen. Beispielsweise können Benutzer dieser Gruppe Objekte auflisten und Objektdaten, Metadaten und Tags lesen. Wenn Sie diese Option auswählen, wird die JSON-Zeichenfolge für eine schreibgeschützte Gruppenrichtlinie im Textfeld angezeigt. Sie können diese Zeichenfolge nicht bearbeiten.
- **Vollzugriff**: Benutzer in dieser Gruppe haben vollen Zugriff auf S3-Ressourcen, einschließlich Buckets. Wenn Sie diese Option auswählen, wird die JSON-Zeichenfolge für eine Gruppenrichtlinie mit vollem Zugriff im Textfeld angezeigt. Sie können diese Zeichenfolge nicht bearbeiten.
- **Ransomware-Minderung**: Diese Beispielrichtlinie gilt für alle Buckets dieses Mandanten. Benutzer in dieser Gruppe können allgemeine Aktionen ausführen, aber keine Objekte dauerhaft aus Buckets löschen, für die die Objektversionierung aktiviert ist.

  Tenant Manager-Benutzer mit der Berechtigung „Alle Buckets verwalten“ können diese Gruppenrichtlinie außer Kraft setzen. Beschränken Sie die Berechtigung „Alle Buckets verwalten“ auf vertrauenswürdige Benutzer und verwenden Sie, sofern verfügbar, die Multi-Faktor-Authentifizierung (MFA).

- **Benutzerdefiniert**: Benutzern in der Gruppe werden die Berechtigungen erteilt, die Sie im Textfeld angeben.

Beispiel: Gruppe vollen Zugriff auf alle Buckets gewähren

In diesem Beispiel wird allen Mitgliedern der Gruppe der vollständige Zugriff auf alle Buckets gewährt, die dem Mandantenkonto gehören, sofern die Bucket-Richtlinie diesen Zugriff nicht ausdrücklich verweigert.

```
{
  "Statement": [
    {
      "Action": "s3:*",
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::*"
    }
  ]
}
```

Beispiel: Gruppe schreibgeschützten Zugriff auf alle Buckets gewähren

In diesem Beispiel haben alle Mitglieder der Gruppe schreibgeschützten Zugriff auf S3-Ressourcen, sofern dies nicht ausdrücklich durch die Bucket-Richtlinie verweigert wird. Beispielsweise können Benutzer dieser Gruppe Objekte auflisten und Objektdaten, Metadaten und Tags lesen.

```
{
  "Statement": [
    {
      "Sid": "AllowGroupReadOnlyAccess",
      "Effect": "Allow",
      "Action": [
        "s3:ListAllMyBuckets",
        "s3:ListBucket",
        "s3:ListBucketVersions",
        "s3:GetObject",
        "s3:GetObjectTagging",
        "s3:GetObjectVersion",
        "s3:GetObjectVersionTagging"
      ],
      "Resource": "arn:aws:s3:::*"
    }
  ]
}
```

Beispiel: Gruppenmitgliedern vollen Zugriff nur auf ihren „Ordner“ in einem Bucket gewähren

In diesem Beispiel dürfen Mitglieder der Gruppe nur ihren spezifischen Ordner (Schlüsselpräfix) im angegebenen Bucket auflisten und darauf zugreifen. Beachten Sie, dass bei der Festlegung des Datenschutzes dieser Ordner Zugriffsberechtigungen aus anderen Gruppenrichtlinien und der Bucket-Richtlinie berücksichtigt werden sollten.

```
{
  "Statement": [
    {
      "Sid": "AllowListBucketOfASpecificUserPrefix",
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::department-bucket",
      "Condition": {
        "StringLike": {
          "s3:prefix": "${aws:username}/*"
        }
      }
    },
    {
      "Sid": "AllowUserSpecificActionsOnlyInTheSpecificUserPrefix",
      "Effect": "Allow",
      "Action": "s3:*Object",
      "Resource": "arn:aws:s3:::department-bucket/${aws:username}/*"
    }
  ]
}
```