Beispiele für Verfahren und APIs

StorageGRID solutions and resources

NetApp
November 21, 2025

Inhalt

Beispiele für Verfahren und APIs

Testen und Demonstrieren der S3 Verschlüsselungsoptionen auf StorageGRID

Von Aron Klein

StorageGRID und die S3-API bieten verschiedene Methoden zur Verschlüsselung von Daten im Ruhezustand. Weitere Informationen finden Sie unter "Prüfen Sie die StorageGRID Verschlüsselungsmethoden".

In diesem Leitfaden werden die S3-API-Verschlüsselungsmethoden demonstriert.

Serverseitige Verschlüsselung (SSE)

Mit SSE kann der Client ein Objekt speichern und mit einem eindeutigen Schlüssel verschlüsseln, der von StorageGRID verwaltet wird. Wenn das Objekt angefordert wird, wird das Objekt durch den in StorageGRID gespeicherten Schlüssel entschlüsselt.

Beispiel: SSE

- SETZEN Sie ein Objekt mit SSE

```
aws s3api put-object --bucket <bucket>  --key <file> --body "<file>"
--server-side-encryption AES256 --endpoint-url https://s3.example.com
```

- LEITEN Sie das Objekt, um die Verschlüsselung zu überprüfen

```
aws s3api head-object --bucket <bucket>  --key <file>   --endpoint-url
https://s3.example.com
```

```
{
    "AcceptRanges": "bytes",
    "LastModified": "2022-05-02T19:03:03+00:00",
    "ContentLength": 47,
    "ETag": "\"82e8bfb872e778a4687a26e6c0b36bc1\"",
    "ContentType": "text/plain",
    "ServerSideEncryption": "AES256",
    "Metadata": {}
}
```

- GET das Objekt

```
aws s3api get-object --bucket <bucket>  --key <file> <file> --endpoint
-url https://s3.example.com
```

Serverseitige Verschlüsselung mit vom Kunden bereitgestellten Schlüsseln (SSE-C)

Mit SSE kann der Client ein Objekt speichern und mit einem eindeutigen Schlüssel verschlüsseln, der vom Client mit dem Objekt bereitgestellt wird. Wenn das Objekt angefordert wird, muss derselbe Schlüssel bereitgestellt werden, um das Objekt zu entschlüsseln und zurückzugeben.

Beispiel SSE-C

- Sie können zu Test- oder Demonstrationszwecken einen Schlüssel erstellen
  - Erstellen eines Verschlüsselungsschlüssels

```
openssl enc -aes-128-cbc -pass pass:secret -P`
```

```
salt=E9DBB6603C7B3D2A
key=23832BAC16516152E560F933F261BF03
iv =71E87C0F6EC3C45921C2754BA131A315
```

- Legen Sie ein Objekt mit dem generierten Schlüssel

```
aws s3api put-object --bucket <bucket> --key <file> --body "file" --sse
-customer-algorithm AES256 --sse-customer-key
23832BAC16516152E560F933F261BF03 --endpoint-url https://s3.example.com
```

- Das Objekt in den Kopf stellen

```
aws s3api head-object --bucket <bucket> --key <file> --sse-customer
-algorithm AES256 --sse-customer-key 23832BAC16516152E560F933F261BF03
--endpoint-url https://s3.example.com
```

```
{
    "AcceptRanges": "bytes",
    "LastModified": "2022-05-02T19:20:02+00:00",
    "ContentLength": 47,
    "ETag": "\"f92ef20ab87e0e13951d9bee862e9f9a\"",
    "ContentType": "binary/octet-stream",
    "Metadata": {},
    "SSECustomerAlgorithm": "AES256",
    "SSECustomerKeyMD5": "rjGuMdjLpPV1eRuotNaPMQ=="
}
```

Wenn Sie den Verschlüsselungsschlüssel nicht angeben, erhalten Sie einen Fehler „ein Fehler ist aufgetreten (404), wenn Sie den HeadObject-Vorgang aufrufen: Nicht gefunden“.

- Get das Objekt

```
aws s3api get-object --bucket <bucket> --key <file> <file> --sse
-customer-algorithm AES256 --sse-customer-key
23832BAC16516152E560F933F261BF03 --endpoint-url https://s3.example.com
```

Wenn Sie den Verschlüsselungsschlüssel nicht bereitstellen, erhalten Sie beim Aufruf des GetObject-Vorgangs einen Fehler „aufgetreten (InvalidRequest): Das Objekt wurde mit einer Form von serverseitiger Verschlüsselung gespeichert. Zum Abrufen des Objekts müssen die richtigen Parameter angegeben werden.“

Bucket-serverseitige Verschlüsselung (SSE-S3)

Mit SSE-S3 kann der Client ein Standardverschlüsselungsverhalten für alle in einem Bucket gespeicherten Objekte definieren. Die Objekte werden mit einem eindeutigen Schlüssel verschlüsselt, der von StorageGRID gemanagt wird. Wenn das Objekt angefordert wird, wird das Objekt von dem in StorageGRID gespeicherten Schlüssel entschlüsselt.

Beispiel für Bucket SSE-S3

- Erstellen eines neuen Buckets und Festlegen einer Standardverschlüsselungsrichtlinie
  - Erstellen eines neuen Buckets

    ```
    aws s3api create-bucket --bucket <bucket> --region us-east-1
    --endpoint-url https://s3.example.com
    ```

  - Put Bucket-Verschlüsselung

```
aws s3api put-bucket-encryption --bucket <bucket> --server-side
-encryption-configuration '{"Rules":
[{"ApplyServerSideEncryptionByDefault": {"SSEAlgorithm":
"AES256"}}]}' --endpoint-url https://s3.example.com
```

- Legen Sie ein Objekt in den Bucket

```
aws s3api put-object --bucket <bucket> --key <file> --body "file"
--endpoint-url https://s3.example.com
```

- Das Objekt in den Kopf stellen

```
aws s3api head-object --bucket <bucket> --key <file> --endpoint-url
https://s3.example.com
```

```
{
"AcceptRanges": "bytes",
"LastModified": "2022-05-02T20:16:23+00:00",
"ContentLength": 47,
"ETag": "\"82e8bfb872e778a4687a26e6c0b36bc1\"",
"ContentType": "binary/octet-stream",
"ServerSideEncryption": "AES256",
"Metadata": {}
}
```

- GET das Objekt

```
aws s3api get-object --bucket <bucket>  --key <file> <file> --endpoint
-url https://s3.example.com
```

S3-Objektsperre auf StorageGRID testen und demonstrieren

Von Aron Klein

Object Lock bietet ein WORM-Modell, um das Löschen oder Überschreiben von Objekten zu verhindern. Die StorageGRID Implementierung von Objektsperrung wird auf Cohasset überprüft, um gesetzliche Vorgaben einzuhalten, den gesetzlichen Aufbewahrungs- und Compliance-Modus für Objektspeicherung zu unterstützen und standardmäßige Bucket-Aufbewahrungsrichtlinien einzuhalten.

In diesem Handbuch wird die S3-Objekt-Lock-API demonstriert.

Gesetzliche Aufbewahrungspflichten

- Object Lock Legal Hold ist ein einfacher ein-/Ausschaltstatus, der auf ein Objekt angewendet wird.

```
aws s3api put-object-legal-hold --bucket <bucket> --key <file> --legal
-hold Status=ON --endpoint-url https://s3.company.com
```

- Überprüfen Sie es mit EINEM GET-Vorgang.

```
aws s3api get-object-legal-hold --bucket <bucket> --key <file>
--endpoint-url https://s3.company.com
```

```
{
    "LegalHold": {
        "Status": "ON"
    }
}
```

- Deaktivieren Sie die gesetzliche Aufbewahrungspflichten

```
aws s3api put-object-legal-hold --bucket <bucket> --key <file> --legal
-hold Status=OFF --endpoint-url https://s3.company.com
```

- Überprüfen Sie es mit EINEM GET-Vorgang.

```
aws s3api get-object-legal-hold --bucket <bucket> --key <file>
--endpoint-url https://s3.company.com
```

```
{
    "LegalHold": {
        "Status": "OFF"
    }
}
```

Compliance-Modus

- Die Objektspeicherung erfolgt mit einer Aufbewahrung bis zum Zeitstempel.

```
aws s3api put-object-retention --bucket <bucket> --key <file>
--retention '{"Mode":"COMPLIANCE", "RetainUntilDate": "2025-06-
10T16:00:00"}' --endpoint-url https://s3.company.com
```

- Überprüfen Sie den Aufbewahrungsstatus

```
aws s3api get-object-retention --bucket <bucket> --key <file> --endpoint
-url https://s3.company.com
+
```

```
{
    "Retention": {
        "Mode": "COMPLIANCE",
        "RetainUntilDate": "2025-06-10T16:00:00+00:00"
    }
}
```

Standardaufbewahrung

- Legen Sie den Aufbewahrungszeitraum in Tagen und Jahren als Aufbewahrungsdatum fest, das mit der API pro Objekt definiert wurde.

```
aws s3api put-object-lock-configuration --bucket <bucket> --object-lock
-configuration '{"ObjectLockEnabled": "Enabled", "Rule": {
"DefaultRetention": { "Mode": "COMPLIANCE", "Days": 10 }}}' --endpoint
-url https://s3.company.com
```

- Überprüfen Sie den Aufbewahrungsstatus

```
aws s3api get-object-lock-configuration --bucket <bucket> --endpoint-url
https://s3.company.com
```

```
{
    "ObjectLockConfiguration": {
        "ObjectLockEnabled": "Enabled",
        "Rule": {
            "DefaultRetention": {
                "Mode": "COMPLIANCE",
                "Days": 10
            }
        }
    }
}
```

- Legen Sie ein Objekt in den Bucket

```
aws s3api put-object --bucket <bucket> --key <file> --body "file"
--endpoint-url https://s3.example.com
```

- Die auf dem Bucket festgelegte Aufbewahrungsdauer wird in einen Aufbewahrungszeitstempel des Objekts konvertiert.

```
aws s3api get-object-retention --bucket <bucket> --key <file> --endpoint
-url https://s3.company.com
```

```
{
    "Retention": {
        "Mode": "COMPLIANCE",
        "RetainUntilDate": "2022-03-02T15:22:47.202000+00:00"
    }
}
```

Testen Löschen eines Objekts mit einer definierten Aufbewahrung

Objekt Lock basiert auf der Versionierung. Die Aufbewahrung ist für eine Version des Objekts definiert. Wenn versucht wird, ein Objekt mit einer definierten Aufbewahrung zu löschen, und keine Version angegeben wird, wird als aktuelle Version des Objekts eine Löschmarkierung erstellt.

- Löschen Sie das Objekt mit definierter Aufbewahrung

```
aws s3api delete-object --bucket <bucket> --key <file> --endpoint-url
https://s3.example.com
```

- Listen Sie die Objekte im Bucket auf

```
aws s3api list-objects --bucket <bucket> --endpoint-url
https://s3.example.com
```

  - Beachten Sie, dass das Objekt nicht aufgeführt ist.
- Listen Sie Versionen auf, um die Löschen-Markierung und die ursprüngliche gesperrte Version anzuzeigen

```
aws s3api list-object-versions --bucket <bucket> --prefix <file>
--endpoint-url https://s3.example.com
```

```
{
    "Versions": [
        {
            "ETag": "\"82e8bfb872e778a4687a26e6c0b36bc1\"",
            "Size": 47,
            "StorageClass": "STANDARD",
            "Key": "file.txt",
            "VersionId":
"RDVDMjYwMTQtQkNDQS0xMUVDLThGOEUtNjQ3NTAwQzAxQTk1",
            "IsLatest": false,
            "LastModified": "2022-04-15T14:46:29.734000+00:00",
            "Owner": {
                "DisplayName": "Tenant01",
                "ID": "56622399308951294926"
            }
        }
    ],
    "DeleteMarkers": [
        {
            "Owner": {
                "DisplayName": "Tenant01",
                "ID": "56622399308951294926"
            },
            "Key": "file01.txt",
            "VersionId":
"QjVDQzgzOTAtQ0FGNi0xMUVDLThFMzgtQ0RGMjAwQjk0MjM1",
            "IsLatest": true,
            "LastModified": "2022-05-03T15:35:50.248000+00:00"
        }
    ]
}
```

- Löschen Sie die gesperrte Version des Objekts

```
aws s3api delete-object  --bucket <bucket> --key <file> --version-id
"<VersionId>" --endpoint-url https://s3.example.com
```

```
An error occurred (AccessDenied) when calling the DeleteObject
operation: Access Denied
```

Richtlinien und Berechtigungen in StorageGRID

Hier sind Beispielrichtlinien und Berechtigungen in StorageGRID S3.

Die Struktur einer Richtlinie

In StorageGRID sind die Gruppenrichtlinien mit den S3 Service-Richtlinien für AWS Benutzer (IAM) identisch.

Gruppenrichtlinien sind in StorageGRID erforderlich. Ein Benutzer mit S3-Zugriffsschlüsseln, aber keiner Benutzergruppe zugewiesen oder einer Gruppe ohne Richtlinie zugewiesen, die einige Berechtigungen erteilt, kann auf keine Daten zugreifen.

Bucket- und Gruppenrichtlinien verwenden die meisten Elemente gemeinsam. Richtlinien werden im json-Format erstellt und können mit dem erstellt werden "AWS-Richtliniengenerator"

Alle Richtlinien definieren den Effekt, die Aktion(en) und die Ressource(en). In Bucket-Richtlinien wird auch ein Principal definiert.

Der **Effekt** wird entweder sein, die Anfrage zuzulassen oder abzulehnen.

Der * Principal*

- Gilt nur für Bucket-Richtlinien.
- Der Hauptbenutzer ist der/die Konto(e)/Benutzer, dem/den die Berechtigungen gewährt oder verweigert werden.
- Kann definiert werden als:
  - Ein Platzhalter „+*“

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

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

  - Eine Mandanten-ID für alle Benutzer in einem Mandanten (entspricht AWS-Konto)

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

- Ein Benutzer (lokal oder föderiert aus dem Mandanten, der sich im Bucket befindet, oder ein anderer Mandant im Grid)

```
"Principal": { "AWS":
"arn:aws:iam::76233906934699427431:user/tenant1user1" }
```

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:federated-
user/tenant2user1" }
```

- Eine Gruppe (lokal oder föderiert aus dem Mandanten, der sich im Bucket befindet, oder ein anderer Mandant im Grid).

```
"Principal": { "AWS":
"arn:aws:iam::76233906934699427431:group/DevOps"  }
```

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

Die **Aktion** ist der Satz von S3-Operationen, die dem/den Benutzer(n) gewährt oder verweigert werden.

> Für Gruppenrichtlinien ist die zulässige Aktion s3:ListBucket erforderlich, damit Benutzer alle S3-Aktionen ausführen können.

Die **Ressource** ist der Eimer oder Eimer, dem die Principals die Fähigkeit zur Ausführung der Aktionen gewährt oder verweigert werden. Optional kann es eine **Bedingung** geben, wenn die Richtlinienaktion gültig ist.

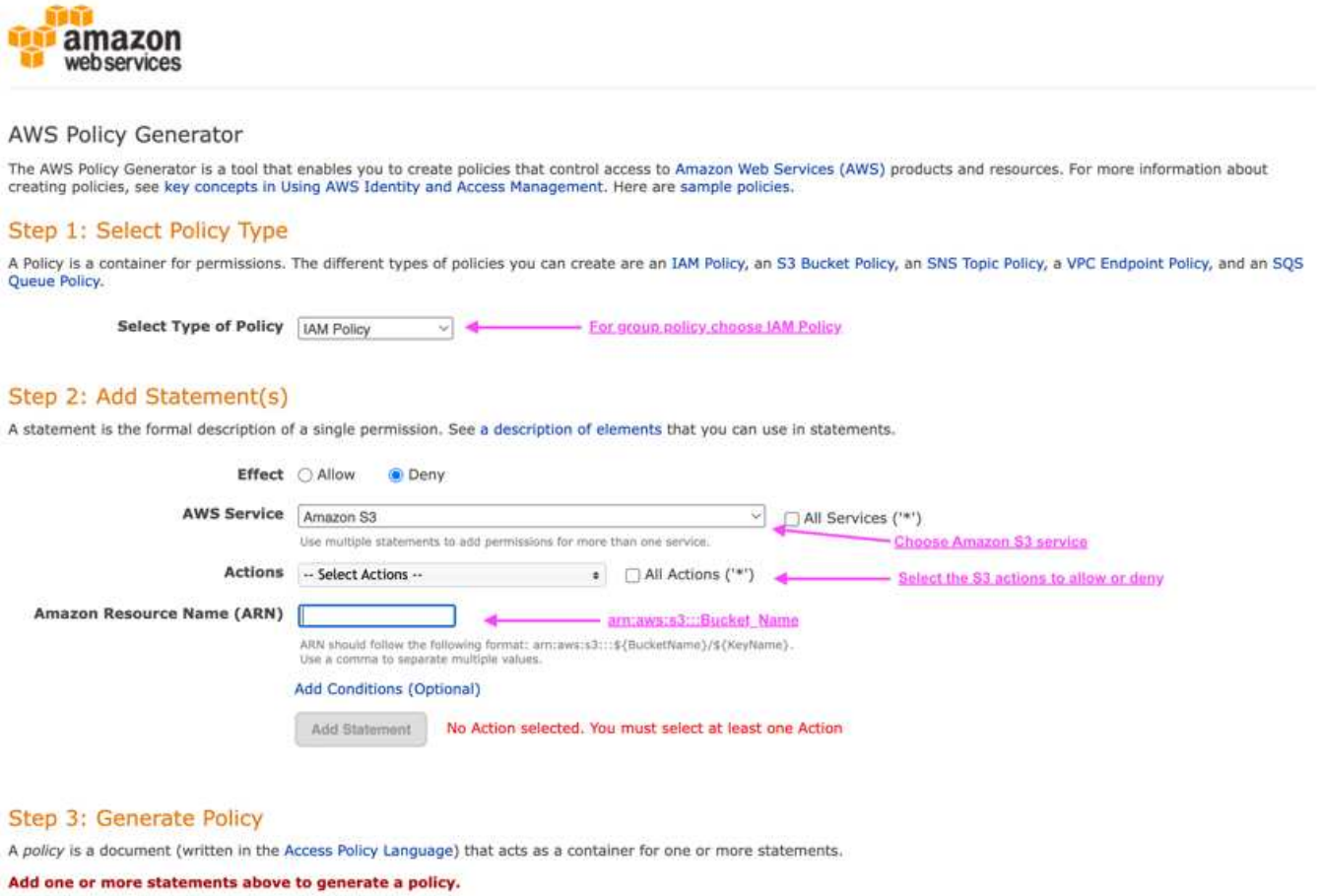
Das Format der JSON-Richtlinie sieht wie folgt aus:

```
{
  "Statement": [
    {
      "Sid": "Custom name for this permission",
      "Effect": "Allow or Deny",
      "Principal": {
        "AWS": [
          "arn:aws:iam::tenant_ID::user/User_Name",
          "arn:aws:iam::tenant_ID::federated-user/User_Name",
          "arn:aws:iam::tenant_ID:group/Group_Name",
          "arn:aws:iam::tenant_ID:federated-group/Group_Name",
          "tenant_ID"
        ]
      },
      "Action": [
        "s3:ListBucket",
        "s3:Other_Action"
      ],
      "Resource": [
        "arn:aws:s3:::Example_Bucket",
        "arn:aws:s3:::Example_Bucket/*"
      ],
    }
  ]
}
```

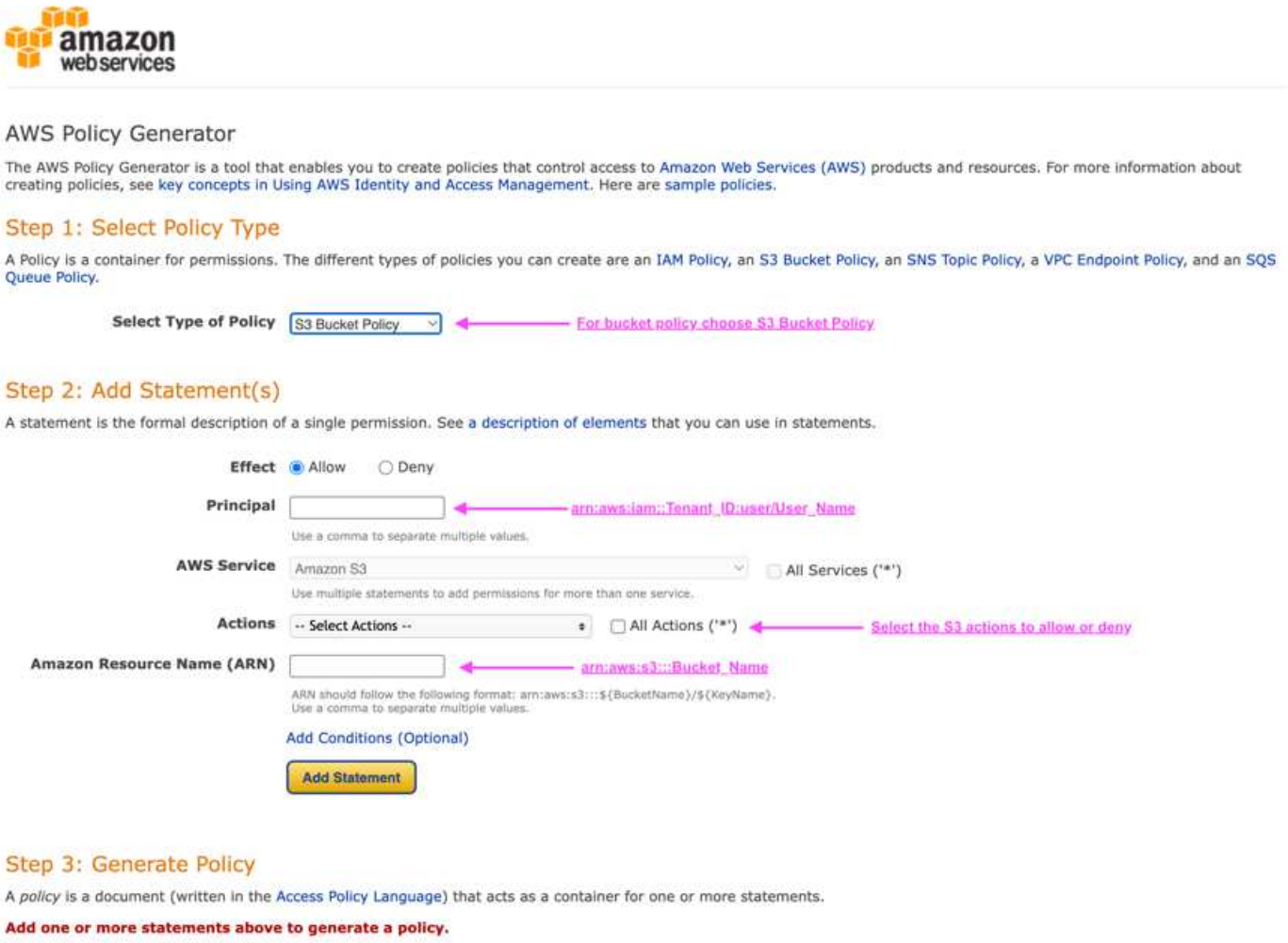
Verwenden des AWS-Richtliniengenerators

Der AWS Richtlinien-Generator ist ein großartiges Werkzeug, um den json-Code mit dem richtigen Format und den Informationen zu erhalten, die Sie zu implementieren versuchen.

So generieren Sie die Berechtigungen für eine StorageGRID-Gruppenrichtlinie: * Wählen Sie die IAM-Richtlinie für den Typ der Richtlinie aus. * Wählen Sie die Schaltfläche für den gewünschten Effekt - Zulassen oder verweigern. Es empfiehlt sich, Ihre Richtlinien mit den Deny-Berechtigungen zu starten und dann die Allow-Berechtigungen * in das Dropdown-Menü Actions hinzuzufügen. Klicken Sie auf das Feld neben so vielen S3-Aktionen, die Sie in diese Berechtigung oder das Feld „All Actions“ aufnehmen möchten. * Geben Sie die Bucket-Pfade in das Feld Amazon Resource Name (ARN) ein. Fügen Sie vor dem Bucket-Namen „arn:aws:s3:::“ ein. Beispiel: „arn:aws:s3::example_bucket“

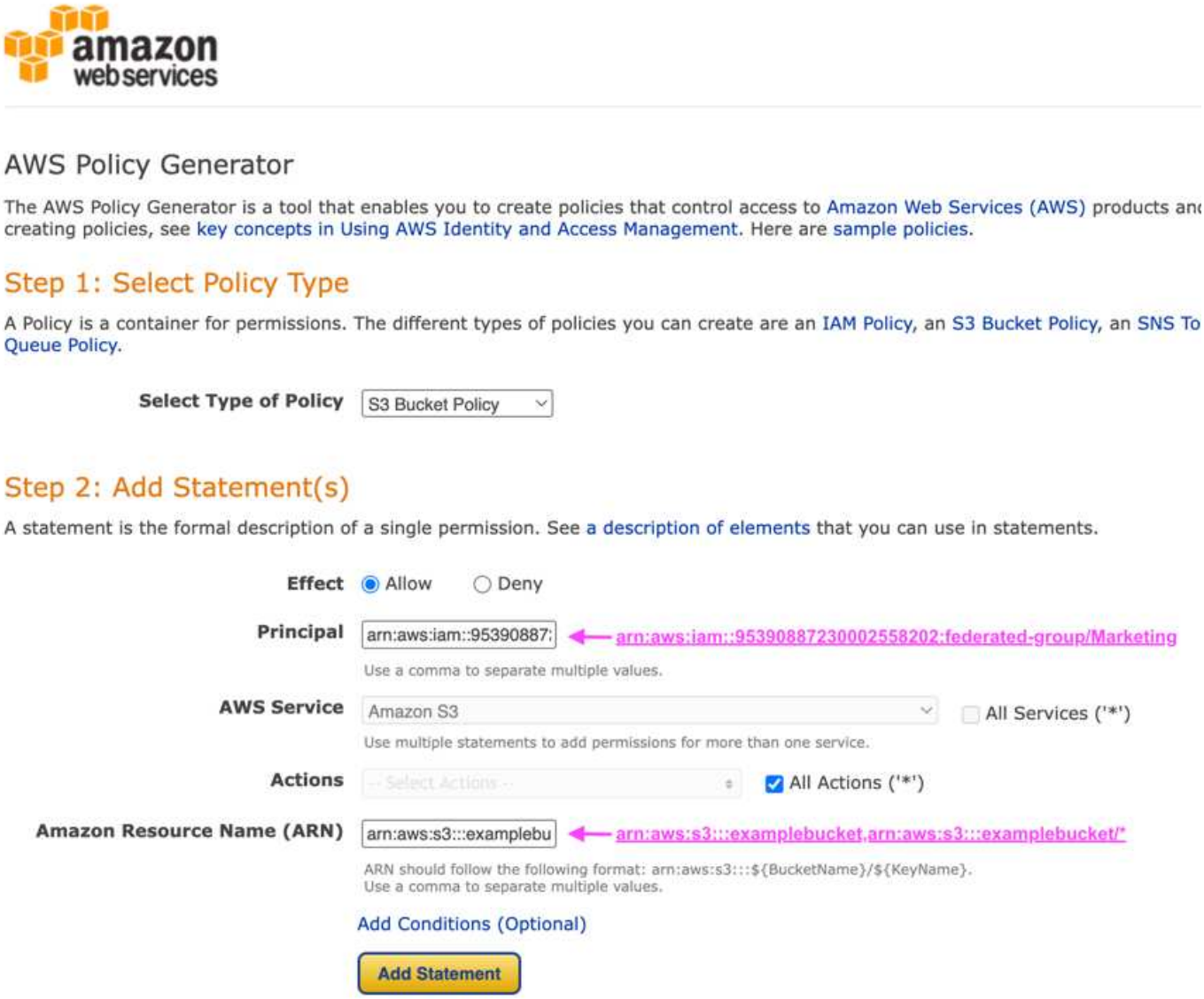


So generieren Sie die Berechtigungen für eine Bucket-Richtlinie: * Wählen Sie die S3-Bucket-Richtlinie für den Typ der Richtlinie aus. * Wählen Sie die Schaltfläche für den gewünschten Effekt - Zulassen oder verweigern. Es empfiehlt sich, Ihre Richtlinien mit den Berechtigungen „verweigern“ zu starten und anschließend den Typ „Berechtigungen zulassen“ * in die Benutzer- oder Gruppeninformationen für den Prinzipal einzufügen. * Klicken Sie in der Dropdown-Liste Aktionen auf das Feld neben so vielen S3-Aktionen, die Sie in diese Berechtigung oder das Feld "Alle Aktionen" aufnehmen möchten. * Geben Sie die Bucket-Pfade in das Feld Amazon Resource Name (ARN) ein. Fügen Sie vor dem Bucket-Namen „arn:aws:s3:::“ ein. Beispiel: „arn:aws:s3::example_bucket“



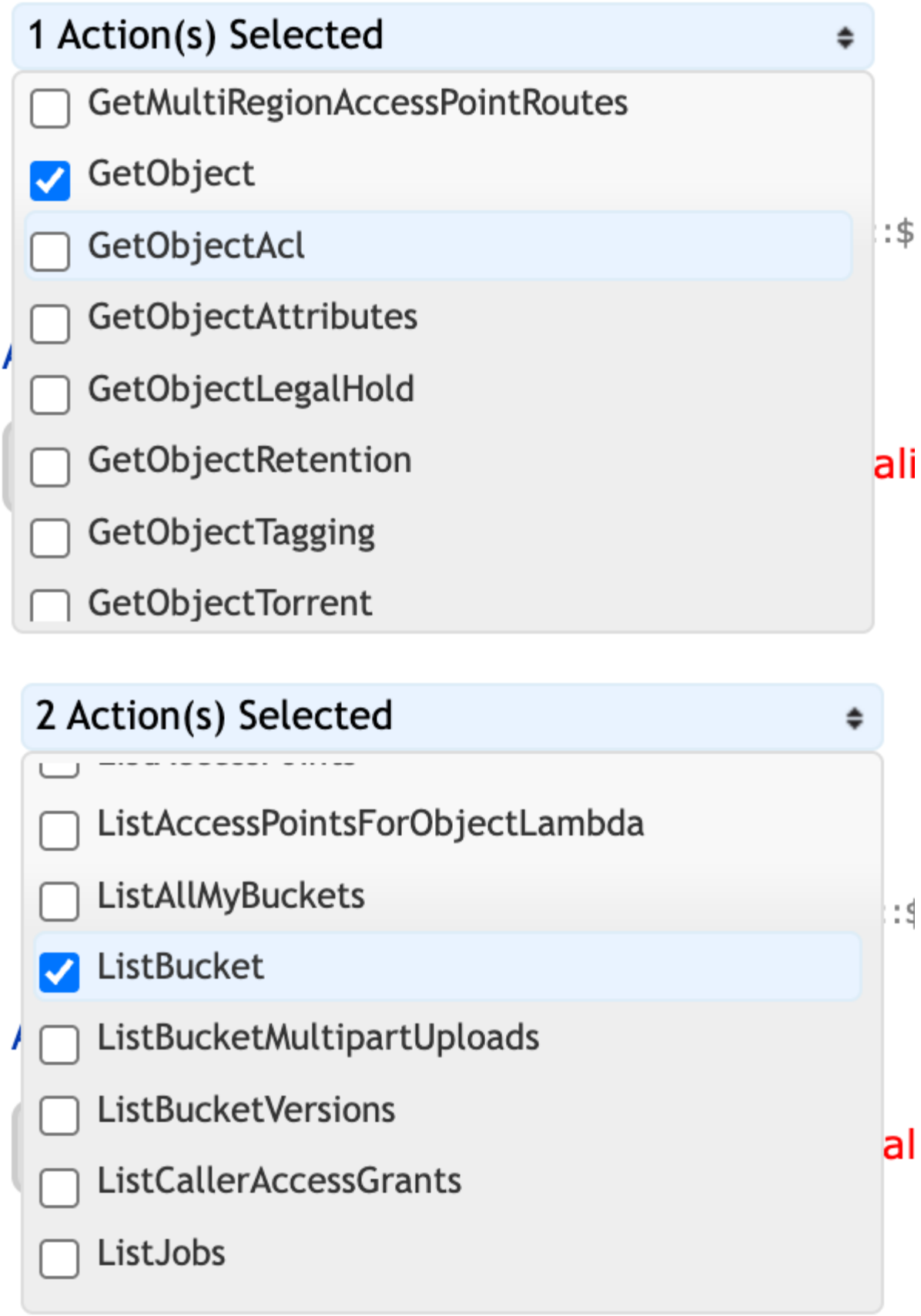
Wenn Sie beispielsweise eine Bucket-Richtlinie erstellen möchten, die allen Benutzern die Ausführung von GetObject-Operationen für alle Objekte im Bucket ermöglicht, während nur Benutzern, die der Gruppe „Marketing“ im angegebenen Konto angehören, Vollzugriff gewährt wird.

- Wählen Sie als Richtlinientyp S3-Bucket-Richtlinie aus.
- Wählen Sie den Zulassen-Effekt
- Geben Sie die Informationen der Marketinggruppe ein - arn:aws:iam::95390887230002558202:Federated-Group/Marketing
- Klicken Sie auf das Feld für „Alle Aktionen“.
- Geben Sie die Bucket-Informationen ein - arn:aws:s3:::example_bucket,arn:aws:s3:::example_bucket/*

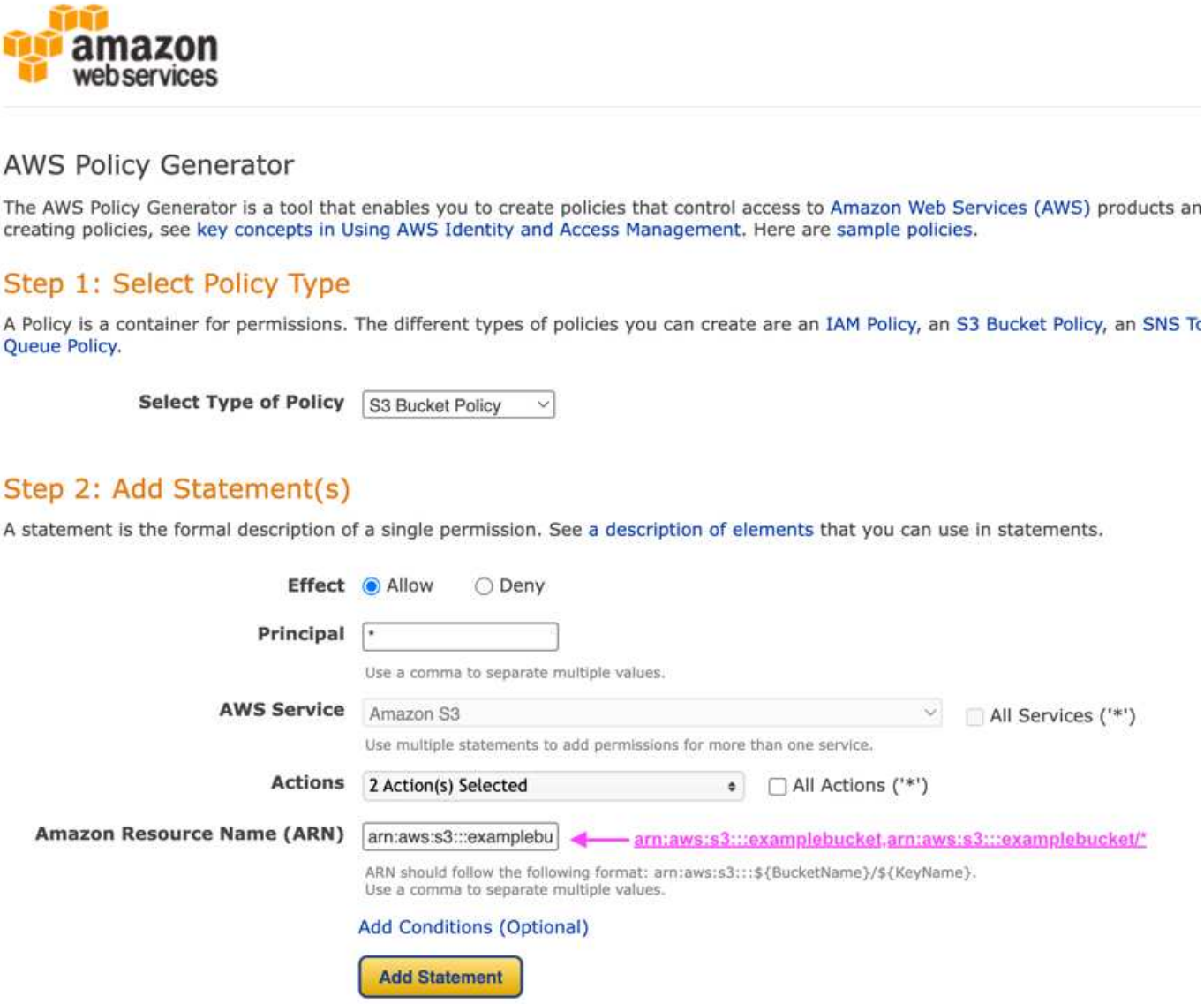


- Klicken Sie auf die Schaltfläche „Anweisung hinzufügen“

- Wählen Sie den Zulassen-Effekt
- Geben Sie das Sternchen +* für alle ein
- Klicken Sie auf das Feld neben GetObject und ListBucket Actions“

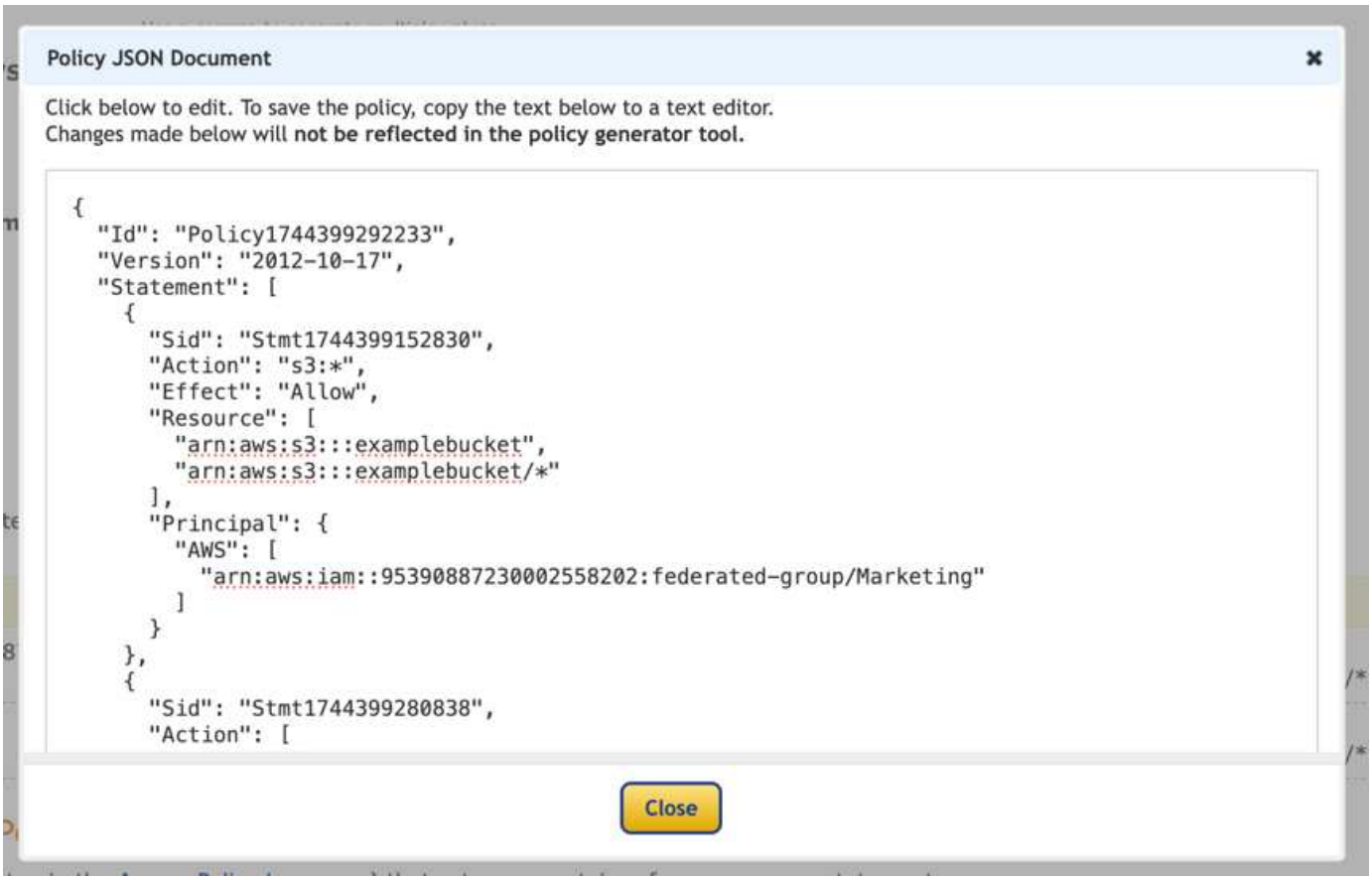


- Geben Sie die Bucket-Informationen ein - arn:aws:s3:::example_bucket,arn:aws:s3:::example_bucket/*



- Klicken Sie auf die Schaltfläche „Anweisung hinzufügen“

- Klicken Sie auf die Schaltfläche „Richtlinie generieren“. Daraufhin wird ein Popup-Fenster mit der erstellten Richtlinie angezeigt.



- Kopieren Sie den vollständigen json-Text, der wie folgt aussehen sollte:

```
{
  "Id": "Policy1744399292233",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1744399152830",
      "Action": "s3:*",
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::example_bucket",
        "arn:aws:s3:::example_bucket/*"
      ],
      "Principal": {
        "AWS": [
          "arn:aws:iam::95390887230002558202:federated-group/Marketing"
        ]
      }
    },
    {
      "Sid": "Stmt1744399280838",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::example_bucket",
        "arn:aws:s3:::example_bucket/*"
      ],
      "Principal": "*"
    }
  ]
}
```

Dieser json kann wie sie ist verwendet werden, oder Sie können die ID- und Versionszeilen über der Zeile "Anweisung" entfernen und Sie können die Sid für jede Berechtigung mit einem aussagekräftigeren Titel für jede Berechtigung anpassen oder diese können auch entfernt werden.

Beispiel:

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

Gruppenrichtlinien (IAM)

Bucket-Zugriff im Home Directory-Stil

Diese Gruppenrichtlinie erlaubt Benutzern nur den Zugriff auf Objekte im Bucket mit dem Namen „username“.

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

  ]
}
```

Erstellung von Bucket-Objektsperrung verweigern

Diese Gruppenrichtlinie schränkt Benutzer am Erstellen eines Buckets ein, für den die Objektsperre für den Bucket aktiviert ist.

> Diese Richtlinie wird in der StorageGRID-Benutzeroberfläche nicht durchgesetzt, sie wird nur durch die S3-API durchgesetzt.

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

Aufbewahrungslimit für Objektsperre

Diese Bucket-Richtlinie beschränkt die Aufbewahrungsdauer der Objektsperre auf maximal 10 Tage

```
{
 "Version":"2012-10-17",
 "Id":"CustSetRetentionLimits",
 "Statement": [
   {
    "Sid":"CustSetRetentionPeriod",
    "Effect":"Deny",
    "Principal":"*",
    "Action": [
      "s3:PutObjectRetention"
    ],
    "Resource":"arn:aws:s3:::testlock-01/*",
    "Condition": {
      "NumericGreaterThan": {
        "s3:object-lock-remaining-retention-days":"10"
      }
    }
   }
  ]
}
```

Benutzer daran hindern, Objekte mit VersionID zu löschen

Diese Gruppenrichtlinie schränkt Benutzer davon ab, versionierte Objekte nach VersionID zu löschen

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

Beschränken Sie eine Gruppe auf ein einzelnes Unterverzeichnis (Präfix) mit Lesezugriff

Diese Richtlinie ermöglicht Mitgliedern der Gruppe schreibgeschützten Zugriff auf ein Unterverzeichnis (Präfix) innerhalb eines Buckets. Der Bucket-Name lautet „Study“ und das Unterverzeichnis lautet „study01“.

```
{
    "Statement": [
        {
            "Sid": "AllowUserToSeeBucketListInTheConsole",
            "Action": [
                "s3:ListAllMyBuckets"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::*"
            ]
        },
        {
            "Sid": "AllowRootAndstudyListingOfBucket",
            "Action": [
                "s3:ListBucket"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3::: study"
```

```
            ],
            "Condition": {
                "StringEquals": {
                    "s3:prefix": [
                        "",
                        "study01/"
                    ],
                    "s3:delimiter": [
                        "/"
                    ]
                }
            }
        },
        {
            "Sid": "AllowListingOfstudy01",
            "Action": [
                "s3:ListBucket"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::study"
            ],
            "Condition": {
                "StringLike": {
                    "s3:prefix": [
                        "study01/*"
                    ]
                }
            }
        },
        {
            "Sid": "AllowAllS3ActionsInstudy01Folder",
            "Effect": "Allow",
            "Action": [
                "s3:Getobject"
            ],
            "Resource": [
                "arn:aws:s3:::study/study01/*"
            ]
        }
    ]
}
```

Bucket-Richtlinien

Bucket auf einzelnen Benutzer mit schreibgeschütztem Zugriff beschränken

Diese Richtlinie erlaubt einem einzelnen Benutzer, schreibgeschützten Zugriff auf einen Bucket zu haben und explizit allen anderen Benutzern den zugriff zu verweigert. Die Gruppierung der Ablehenserklärungen an der Spitze der Richtlinie ist eine gute Methode für eine schnellere Bewertung.

```
{
    "Statement": [
        {
            "Sid": "Deny non user1",
            "Effect": "Deny",
            "NotPrincipal": {
                "AWS": "arn:aws:iam::34921514133002833665:user/user1"
            },
            "Action": [
                "s3:*"
            ],
            "Resource": [
                "arn:aws:s3:::bucket1",
                "arn:aws:s3:::bucket1/*"
            ]
        },
        {
            "Sid": "Allow user1 read access to bucket bucket1",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::34921514133002833665:user/user1"
            },
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::bucket1",
                "arn:aws:s3:::bucket1/*"
            ]
        }
    ]
}
```

Beschränken Sie einen Bucket auf einige Benutzer mit schreibgeschütztem Zugriff.

```
{
    "Statement": [
      {
        "Sid": "Deny all S3 actions to employees 002-005",
        "Effect": "deny",
        "Principal": {
          "AWS": [
            "arn:aws:iam::465215141330027038882:user/employee-002",
            "arn:aws:iam::465215141330027038882:user/employee-003",
            "arn:aws:iam::465215141330027038882:user/employee-004",
            "arn:aws:iam::465215141330027038882:user/employee-005"
          ]
        },
        "Action": "*",
        "Resource": [
          "arn:aws:s3:::databucket1",
          "arn:aws:s3:::databucket1/*"
        ]
      },
      {
        "Sid": "Allow read-only access for employees 002-005",
        "Effect": "Allow",
        "Principal": {
          "AWS": [
            "arn:aws:iam::465215141330027038882:user/employee-002",
            "arn:aws:iam::465215141330027038882:user/employee-003",
            "arn:aws:iam::465215141330027038882:user/employee-004",
            "arn:aws:iam::465215141330027038882:user/employee-005"
          ]
        },
        "Action": [
          "s3:GetObject",
          "s3:GetObjectTagging",
          "s3:GetObjectVersion"
        ],
        "Resource": [
          "arn:aws:s3:::databucket1",
          "arn:aws:s3:::databucket1/*"
        ]
      }
    ]
}
```

Beschränken Sie das Löschen versionierter Objekte in einem Bucket

Diese Bucket-Richtlinie beschränkt das Löschen versionierter Objekte durch einen Benutzer (identifiziert durch Benutzer-ID „56622399308951294926“) nach VersionID

```
{
  "Statement": [
    {
      "Action": [
        "s3:DeleteObjectVersion"
      ],
      "Effect": "Deny",
      "Resource": "arn:aws:s3:::verdeny/*",
      "Principal": {
        "AWS": [
          "56622399308951294926"
        ]
      }
    },
    {
      "Action": "s3:*",
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::verdeny/*",
      "Principal": {
        "AWS": [
          "56622399308951294926"
        ]
      }
    }
  ]
}
```

Bucket-Lebenszyklus in StorageGRID

Sie können eine S3-Lebenszyklukonfiguration erstellen, um zu steuern, wann bestimmte Objekte aus dem StorageGRID System gelöscht werden.

Was ist eine Lebenszykluskonfiguration?

Eine Lifecycle-Konfiguration ist ein Satz von Regeln, die auf die Objekte in bestimmten S3-Buckets angewendet werden. Jede Regel gibt an, welche Objekte betroffen sind und wann diese Objekte ablaufen (an einem bestimmten Datum oder nach einigen Tagen).

Jedes Objekt folgt den Aufbewahrungseinstellungen eines S3 Bucket-Lebenszyklus oder einer ILM-Richtlinie. Wenn ein S3-Bucket-Lebenszyklus konfiguriert ist, überschreiben die Lifecycle-Ablaufaktionen die ILM-Richtlinie für Objekte, die mit dem Bucket-Lifecycle-Filter übereinstimmen. Objekte, die nicht mit dem Bucket-Lebenszyklusfilter übereinstimmen, verwenden die Aufbewahrungseinstellungen der ILM-Richtlinie. Wenn ein

Objekt mit einem Bucket-Lebenszyklusfilter übereinstimmt und keine Ablaufaktionen explizit angegeben werden, werden die Aufbewahrungseinstellungen der ILM-Richtlinie nicht verwendet, und es wird impliziert, dass Objektversionen für immer aufbewahrt werden.

Aus diesem Grund kann ein Objekt aus dem Grid entfernt werden, obwohl die Speicheranweisungen in einer ILM-Regel noch auf das Objekt gelten. Oder ein Objekt kann auf dem Raster verbleiben, selbst wenn alle ILM-Platzierungsanweisungen für das Objekt abgelaufen sind.

StorageGRID unterstützt in einer Lebenszykluskonfiguration bis zu 1,000 Lebenszyklusregeln. Jede Regel kann die folgenden XML-Elemente enthalten:

- Ablauf: Löschen eines Objekts, wenn ein bestimmtes Datum erreicht wird oder wenn eine bestimmte Anzahl von Tagen erreicht wird, beginnend mit dem Zeitpunkt der Aufnahme des Objekts.
- NoncurrentVersionExpiration: Löschen Sie ein Objekt, wenn eine bestimmte Anzahl von Tagen erreicht wird, beginnend ab dem Zeitpunkt, an dem das Objekt nicht mehr aktuell wurde.
- Filter (Präfix, Tag)
- Status *ID

StorageGRID unterstützt den Einsatz der folgenden Bucket-Operationen zum Management der Lebenszykluskonfigurationen:

- DeleteBucketLifecycle
- GetBucketLifecycleKonfiguration
- PutBucketLifecycleKonfiguration

Aufbau einer Lifecycle-Policy

Als erster Schritt beim Erstellen einer Lebenszykluskonfiguration erstellen Sie eine JSON-Datei mit einem oder mehreren Regeln. Diese JSON-Datei enthält beispielsweise drei Regeln:

1. **Regel 1** gilt nur für Objekte, die dem Präfix „category1/“ entsprechen und den Wert „key2“ von „tag2“ haben. Der Parameter „Expiration“ gibt an, dass Objekte, die dem Filter entsprechen, am 22. August 2020 um Mitternacht ablaufen.
2. **Regel 2** gilt nur für Objekte, die dem Präfix „category2/“ entsprechen. Der Parameter „Expiration“ gibt an, dass Objekte, die dem Filter entsprechen, 100 Tage nach ihrer Aufnahme ablaufen.

   > Regeln, die eine Anzahl von Tagen angeben, sind relativ zu dem Zeitpunkt, an dem das Objekt aufgenommen wurde. Wenn das aktuelle Datum das Aufnahmedatum plus die Anzahl der Tage überschreitet, werden einige Objekte möglicherweise aus dem Bucket entfernt, sobald die Lebenszykluskonfiguration angewendet wird.

3. **Regel 3** gilt nur für Objekte mit dem Präfix „category3/“. Der Parameter „Expiration“ gibt an, dass alle nicht aktuellen Versionen übereinstimmender Objekte 50 Tage nach ihrer Nicht-Aktualisierung ablaufen.

```
{
    "Rules": [
        {
            "ID": "rule1",
            "Filter": {
                "And": {
                    "Prefix": "category1/",
                    "Tags": [
                        {
                            "Key": "key2",
                            "Value": "tag2"
                        }
                    ]
                }
            },
            "Expiration": {
                "Date": "2020-08-22T00:00:00Z"
            },
            "Status": "Enabled"
        },
        {
            "ID": "rule2",
            "Filter": {
                "Prefix": "category2/"
            },
            "Expiration": {
                "Days": 100
            },
            "Status": "Enabled"
        },
        {
            "ID": "rule3",
            "Filter": {
                "Prefix": "category3/"
            },
            "NoncurrentVersionExpiration": {
                "NoncurrentDays": 50
            },
            "Status": "Enabled"
        }
    ]
}
```

Lifecycle-Konfiguration auf Bucket anwenden

Nachdem Sie die Lebenszykluskonfigurationsdatei erstellt haben, wenden Sie sie auf einen Bucket an, indem Sie eine Anforderung von PutkBucketLifecycleConfiguration ausgeben.

Diese Anforderung wendet die Lebenszykluskonfiguration in der Beispieldatei auf Objekte in einem Bucket mit dem Namen `testbucket`an.

```
aws s3api --endpoint-url <StorageGRID endpoint> put-bucket-lifecycle-
configuration
--bucket testbucket --lifecycle-configuration file://bktjson.json
```

Um zu überprüfen, ob eine Lebenszykluskonfiguration erfolgreich auf den Bucket angewendet wurde, geben Sie eine GetBucketLifecycleConfiguration-Anforderung aus. Beispiel:

```
aws s3api --endpoint-url <StorageGRID endpoint> get-bucket-lifecycle-
configuration
 --bucket testbucket
```

Beispiel-Lebenszyklusrichtlinien für Standard-Buckets (ohne Versionsangabe)

Objekte nach 90 Tagen löschen

Anwendungsfall: Diese Richtlinie eignet sich ideal für die Verwaltung zeitlich begrenzt relevanter Daten, wie z. B. temporäre Dateien, Protokolle oder Zwischenverarbeitungsdaten. Vorteil: Reduzieren Sie die Speicherkosten und sorgen Sie für einen übersichtlichen Bucket.

```
{
    "Rules": [
      {
        "ID": "Delete after 90 day rule",
        "Filter": {},
        "Status": "Enabled",
          "Expiration": {
              "Days": 90
        }
      }
    ]
}
```

Beispiel-Lebenszyklusrichtlinien für versionierte Buckets

Nicht aktuelle Versionen nach 10 Tagen löschen

Anwendungsfall: Diese Richtlinie hilft bei der Verwaltung der Speicherung veralteter Versionsobjekte, die sich im Laufe der Zeit ansammeln und viel Speicherplatz beanspruchen können. Vorteil: Optimieren Sie die

Speichernutzung, indem Sie nur die neueste Version speichern.

```
{
    "Rules": [
            {
        "ID": "NoncurrentVersionExpiration 10 day rule",
        "Filter": {},
        "Status": "Enabled",
          "NoncurrentVersionExpiration": {
              "NoncurrentDays": 10
        }
    }
    ]
}
```

Behalten Sie 5 nicht aktuelle Versionen

Anwendungsfall: Nützlich, wenn Sie eine begrenzte Anzahl früherer Versionen zu Wiederherstellungs- oder Prüfzwecken behalten möchten. Vorteil: Behalten Sie genügend nicht aktuelle Versionen, um ausreichend Verlauf und Wiederherstellungspunkte sicherzustellen.

```
{
    "Rules": [
      {
        "ID": "NewerNoncurrentVersions 5 version rule",
        "Filter": {},
        "Status": "Enabled",
        "NoncurrentVersionExpiration": {
            "NewerNoncurrentVersions": 5
        }
    }
    ]
}
```

Löschmarkierungen entfernen, wenn keine anderen Versionen vorhanden sind

Anwendungsfall: Diese Richtlinie hilft bei der Verwaltung der Löschmarkierungen, die nach dem Entfernen aller nicht aktuellen Versionen übrig bleiben und sich im Laufe der Zeit ansammeln können. Vorteil: Reduzierung unnötiger Unordnung.

```
{
    "Rules": [
    {
        "ID": "Delete marker cleanup rule",
        "Filter": {},
        "Status": "Enabled",
        "Expiration": {
        "ExpiredObjectDeleteMarker": true
        }
    }
    ]
}
```

Löschen Sie aktuelle Versionen nach 30 Tagen, löschen Sie nicht aktuelle Versionen nach 60 Tagen und entfernen Sie die Löschmarkierungen, die durch das Löschen der aktuellen Version erstellt wurden, sobald keine anderen Versionen mehr vorhanden sind.

Anwendungsfall: Bereitstellung eines vollständigen Lebenszyklus für aktuelle und nicht aktuelle Versionen inklusive Löschmarkierungen. Vorteil: Reduzieren Sie die Speicherkosten und sorgen Sie für einen übersichtlichen Bucket, während ausreichend Wiederherstellungspunkte und Verlauf erhalten bleiben.

```
{
  "Rules": [
    {
      "ID": "Delete current version",
      "Filter": {},
      "Status": "Enabled",
      "Expiration": {
        "Days": 30
      }
    },
    {
      "ID": "noncurrent version retention",
      "Filter": {},
      "Status": "Enabled",
      "NoncurrentVersionExpiration": {
        "NoncurrentDays": 60
      }
    },
    {
      "ID": "Markers",
      "Filter": {},
      "Status": "Enabled",
      "Expiration": {
        "ExpiredObjectDeleteMarker": true
      }
    }
  ]
}
```

Entfernen Sie Löschmarkierungen, die keine anderen Versionen haben. Behalten Sie 4 nicht aktuelle Versionen und mindestens 30 Tage Verlauf für Objekte mit dem Präfix „accounts_“ und behalten Sie 2 Versionen und mindestens 10 Tage Verlauf für alle anderen Objektversionen.

Anwendungsfall: Definieren Sie eindeutige Regeln für bestimmte Objekte neben anderen Objekten, um den gesamten Lebenszyklus aktueller und nicht aktueller Versionen inklusive Löschmarkierungen zu verwalten. Vorteil: Reduzieren Sie die Speicherkosten und sorgen Sie für einen übersichtlichen Bucket. Gleichzeitig bleiben ausreichend Wiederherstellungspunkte und Verlaufsdaten erhalten, um verschiedene Kundenanforderungen zu erfüllen.

```
{
  "Rules": [
    {
      "ID": "Markers",
      "Filter": {},
      "Status": "Enabled",
      "Expiration": {
        "ExpiredObjectDeleteMarker": true
      }
    },
    {
      "ID": "accounts version retention",
      "Filter": {"Prefix":"account_"},
      "Status": "Enabled",
      "NoncurrentVersionExpiration": {
        "NewerNoncurrentVersions": 4,
        "NoncurrentDays": 30
      }
    },
    {
      "ID": "noncurrent version retention",
      "Filter": {},
      "Status": "Enabled",
      "NoncurrentVersionExpiration": {
        "NewerNoncurrentVersions": 2,
        "NoncurrentDays": 10
      }
    }
  ]
}
```

Schlussfolgerung

- Überprüfen und aktualisieren Sie Lebenszyklusrichtlinien regelmäßig und richten Sie sie an den ILM- und Datenverwaltungszielen aus.
- Testen Sie Richtlinien in einer Nicht-Produktionsumgebung oder einem Bucket, bevor Sie sie allgemein anwenden, um sicherzustellen, dass sie wie vorgesehen funktionieren.
- Verwenden Sie beschreibende IDs für Regeln, um sie intuitiver zu gestalten, da die logische Struktur komplex werden kann
- Überwachen Sie die Auswirkungen dieser Bucket-Lebenszyklusrichtlinien auf die Speichernutzung und Leistung, um die erforderlichen Anpassungen vorzunehmen.

Copyright-Informationen

Markeninformationen