Procedure ed esempi di API

StorageGRID solutions and resources

NetApp
November 21, 2025

Sommario

Procedure ed esempi di API . . . 1
Testare e dimostrare le opzioni di crittografia S3 su StorageGRID . . . 1
Server Side Encryption (SSE) . . . 1
Crittografia lato server con chiavi fornite dal cliente (SSE-C) . . . 2
Crittografia lato server bucket (SSE-S3) . . . 3
Testare e dimostrare il blocco di oggetti S3 su StorageGRID . . . 4
Conservazione a fini giudiziari . . . 5
Modalità compliance . . . 5
Conservazione predefinita . . . 6
Verificare l’eliminazione di un oggetto con una conservazione definita . . . 7
Criteri e autorizzazioni in StorageGRID . . . 9
La struttura di una politica . . . 9
Utilizzo del generatore di policy AWS . . . 11
Policy di gruppo (IAM) . . . 19
Criteri benna . . . 24
Ciclo di vita del bucket in StorageGRID . . . 26
Che cos’è una configurazione del ciclo di vita . . . 26
Struttura di una politica del ciclo di vita . . . 27
Applica la configurazione del ciclo di vita al bucket . . . 29
Esempi di policy del ciclo di vita per bucket standard (senza controllo di versione) . . . 29
Esempi di policy del ciclo di vita per bucket con versione . . . 29
Conclusione . . . 33

Procedure ed esempi di API

Testare e dimostrare le opzioni di crittografia S3 su StorageGRID

Di Aron Klein

StorageGRID e l'API S3 offrono diversi modi per crittografare i dati inattivi. Per ulteriori informazioni, vedere "Esaminare i metodi di crittografia StorageGRID".

Questa guida illustra i metodi di crittografia dell'API S3.

Server Side Encryption (SSE)

SSE consente al client di memorizzare un oggetto e di crittografarlo con una chiave univoca gestita da StorageGRID. Quando l'oggetto viene richiesto, l'oggetto viene decrittografato dalla chiave memorizzata in StorageGRID.

Esempio SSE

- METTI un oggetto con SSE

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

- TESTA l'oggetto per verificare la crittografia

```
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": {}
}
```

- OTTIENI l'oggetto

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

Crittografia lato server con chiavi fornite dal cliente (SSE-C)

SSE consente al client di memorizzare un oggetto e di crittografarlo con una chiave univoca fornita dal client con l'oggetto. Quando l'oggetto viene richiesto, è necessario fornire la stessa chiave per decrittare e restituire l'oggetto.

Esempio SSE-C.

- A scopo di test o dimostrazione, è possibile creare una chiave di crittografia
  - Creare una chiave di crittografia

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

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

- Inserire un oggetto con la chiave generata

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

- Testa l'oggetto

```
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=="
}
```

Se non si fornisce la chiave di crittografia, viene visualizzato il messaggio di errore "si è verificato un errore (404) durante la chiamata dell'operazione HeadObject: Non trovata"

- Ottieni l'oggetto

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

Se non si fornisce la chiave di crittografia, viene visualizzato l'errore "si è verificato un errore (InvalidRequest) durante la chiamata dell'operazione GetObject: L'oggetto è stato memorizzato utilizzando un modulo di Server Side Encryption. Per recuperare l'oggetto, è necessario fornire i parametri corretti."

Crittografia lato server bucket (SSE-S3)

SSE-S3 consente al client di definire un comportamento di crittografia predefinito per tutti gli oggetti memorizzati in un bucket. Gli oggetti vengono crittografati con una chiave univoca gestita da StorageGRID. Quando l'oggetto viene richiesto, l'oggetto viene decrittografato dalla chiave memorizzata in StorageGRID.

Esempio di bucket SSE-S3

- Creare un nuovo bucket e impostare una policy di crittografia predefinita
  - Creare un nuovo bucket

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

  - Metti la crittografia bucket

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

- Metti un oggetto nel bucket

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

- Testa l'oggetto

```
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": {}
}
```

- OTTIENI l'oggetto

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

Testare e dimostrare il blocco di oggetti S3 su StorageGRID

Di Aron Klein

Object Lock fornisce un modello WORM per impedire l'eliminazione o la sovrascrittura degli oggetti. L'implementazione StorageGRID del blocco degli oggetti viene valutata da Cohasset per soddisfare i requisiti normativi, supportare la modalità di conservazione legale e di conformità per la conservazione degli oggetti e le policy di conservazione predefinite dei bucket.

Questa guida illustra l'API S3 Object Lock.

Conservazione a fini giudiziari

- Object Lock legal hold è un semplice stato on/off applicato a un oggetto.

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

- Verificarlo con un'operazione GET.

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

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

- Disattivare la sospensione legale

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

- Verificarlo con un'operazione GET.

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

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

Modalità compliance

- La conservazione degli oggetti viene eseguita con un periodo di conservazione fino a data e ora.

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

- Verificare lo stato di conservazione

```
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"
    }
}
```

Conservazione predefinita

- Impostare il periodo di conservazione in giorni e anni rispetto alla data di conservazione definita con l'api per oggetto.

```
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
```

- Verificare lo stato di conservazione

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

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

- Metti un oggetto nel bucket

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

- La durata di conservazione impostata sul bucket viene convertita in un indicatore data e ora di conservazione sull'oggetto.

```
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"
    }
}
```

Verificare l'eliminazione di un oggetto con una conservazione definita

Il blocco degli oggetti si basa sul controllo delle versioni. La conservazione viene definita su una versione dell'oggetto. Se si tenta di eliminare un oggetto con una conservazione definita e non viene specificata alcuna versione, viene creato un indicatore di eliminazione come versione corrente dell'oggetto.

- Eliminare l'oggetto con la conservazione definita

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

- Elencare gli oggetti nel bucket

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

  - Notare che l'oggetto non è elencato.

- Elencare le versioni per visualizzare il marker di eliminazione e la versione originale bloccata

```
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"
        }
    ]
}
```

- Eliminare la versione bloccata dell'oggetto

```
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
```

Criteri e autorizzazioni in StorageGRID

Di seguito sono riportati alcuni criteri e autorizzazioni di esempio in StorageGRID S3.

La struttura di una politica

In StorageGRID, le policy di gruppo sono le stesse delle policy di servizio dell'utente AWS (IAM) S3.

I criteri di gruppo sono necessari in StorageGRID. Un utente con S3 chiavi di accesso ma non assegnato a un gruppo di utenti o assegnato a un gruppo senza un criterio che concede alcune autorizzazioni, non potrà accedere a nessun dato.

I criteri bucket e gruppo condividono la maggior parte degli stessi elementi. Le policy vengono create in formato json e possono essere generate utilizzando "Generatore di policy AWS"

Tutti i criteri definiscono l'effetto, le azioni e le risorse. I criteri bucket definiranno anche un'entità.

Il **effetto** sarà quello di consentire o negare la richiesta.

Il **Principal**

- Valido solo per le politiche bucket.
- L'entità è costituita dagli account/utenti ai quali vengono concesse o negate le autorizzazioni.
- Può essere definito come:
  - Un carattere jolly "*"

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

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

  - Un ID tenant per tutti gli utenti in un tenant (equivalente all'account AWS)

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

- Un utente (locale o federato dall'interno del tenant in cui risiede il bucket o un altro tenant nella griglia)

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

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

- Un gruppo (locale o federato dall'interno del tenant in cui risiede il bucket o un altro tenant nella griglia).

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

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

La **azione** è l'insieme di operazioni S3 concesse o negate agli utenti.

Per i criteri di gruppo, l'azione S3:ListBucket Allowed è necessaria per consentire agli utenti di eseguire qualsiasi azione S3.

La **risorsa** è il bucket o i bucket a cui sono stati concessi o negati i principi di eseguire le azioni su. Facoltativamente, può essere presente una condizione * per quando l'azione del criterio è valida.

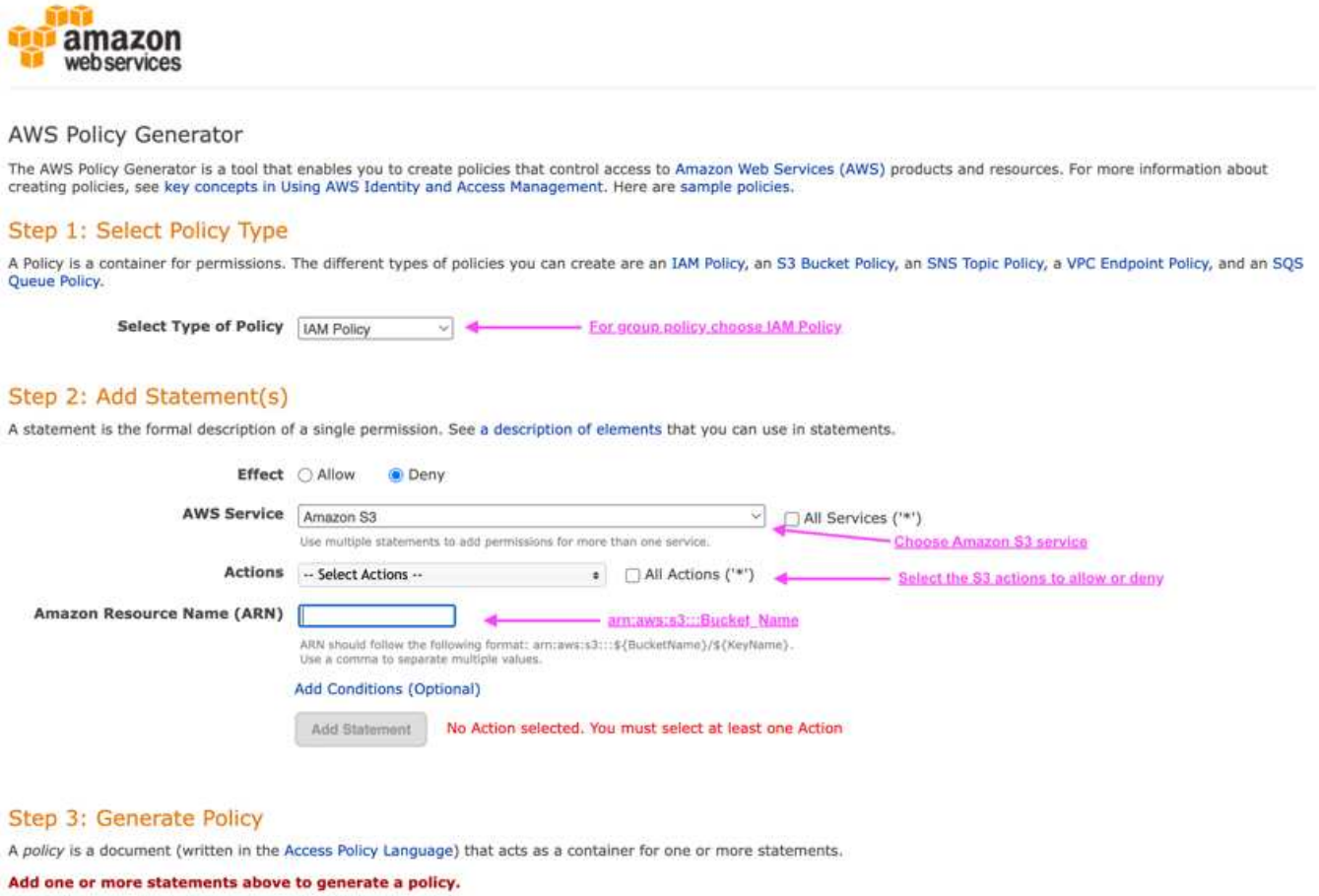
Il formato del criterio JSON sarà il seguente:

```
{
  "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/*"
      ],
    }
  ]
}
```

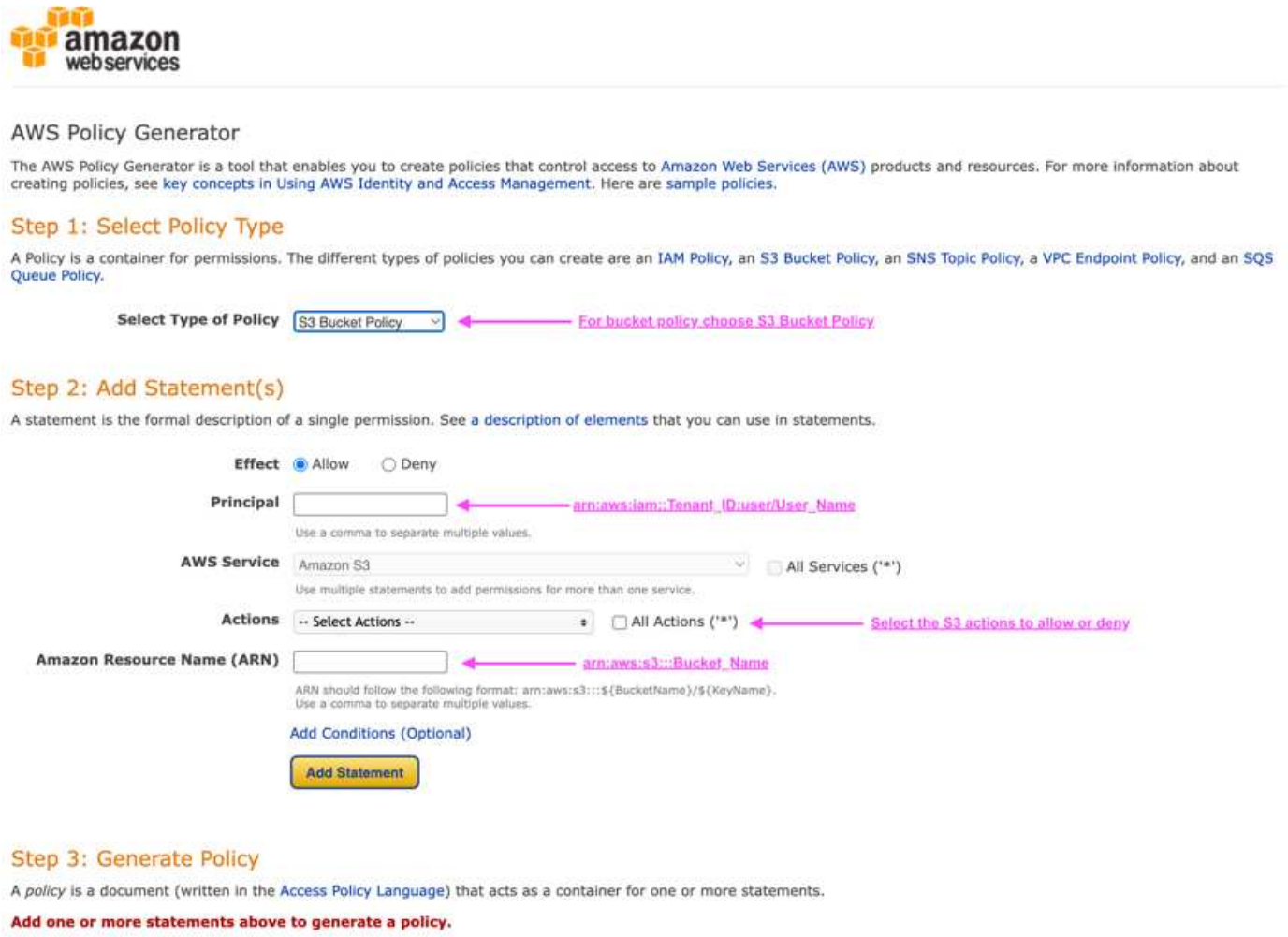
Utilizzo del generatore di policy AWS

AWS Policy generator è un ottimo tool per aiutare a ottenere il codice json con il formato e le informazioni corrette che stai cercando di implementare.

Per generare le autorizzazioni per un criterio di gruppo StorageGRID: * Scegliere il criterio IAM per il tipo di criterio. * Selezionare il pulsante per l'effetto desiderato - Allow (Consenti) o Deny (Nega). È buona norma avviare i criteri con le autorizzazioni di negazione e quindi aggiungere le autorizzazioni di autorizzazione * nel menu a discesa azioni fare clic sulla casella accanto a tutte le S3 azioni che si desidera includere in questa autorizzazione o nella casella "tutte le azioni". * Digitare i percorsi bucket nella casella Amazon Resource Name (ARN). Includere "arn:aws:S3:::" prima del nome bucket. Es. "arn:aws:s3:::example_bucket"

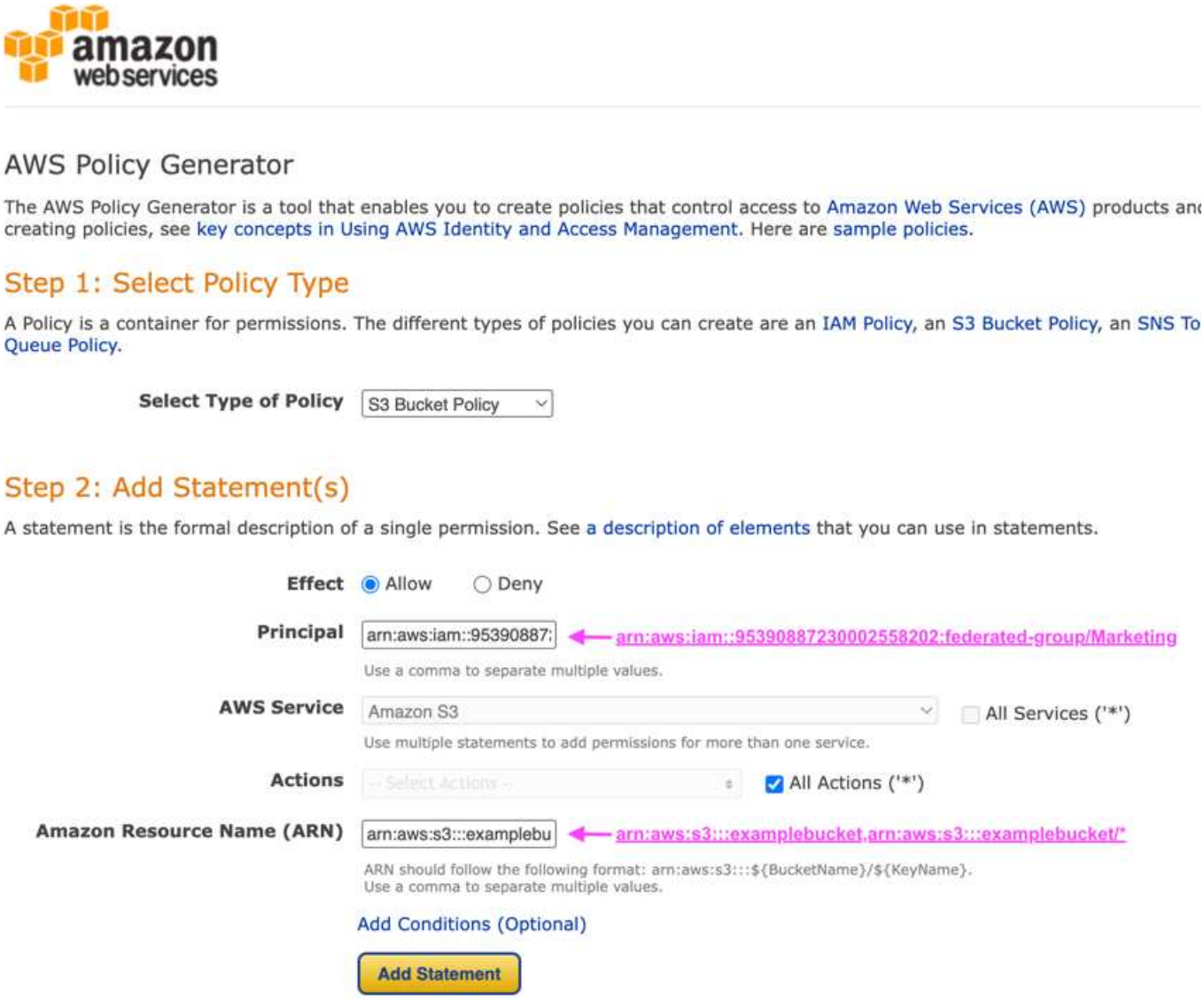


Per generare le autorizzazioni per un criterio bucket: * Scegliere il criterio bucket S3 per il tipo di criterio. * Selezionare il pulsante per l'effetto desiderato - Allow (Consenti) o Deny (Nega). È buona norma avviare i criteri con le autorizzazioni di negazione e quindi aggiungere il tipo Consenti autorizzazioni * nelle informazioni relative all'utente o al gruppo per il principale. * Nell'elenco a discesa azioni, fare clic sulla casella accanto a tutte le S3 azioni che si desidera includere in questa autorizzazione o nella casella "tutte le azioni". * Digitare i percorsi bucket nella casella Amazon Resource Name (ARN). Includere "arn:aws:S3:::" prima del nome bucket. Es. "arn:aws:s3:::example_bucket"



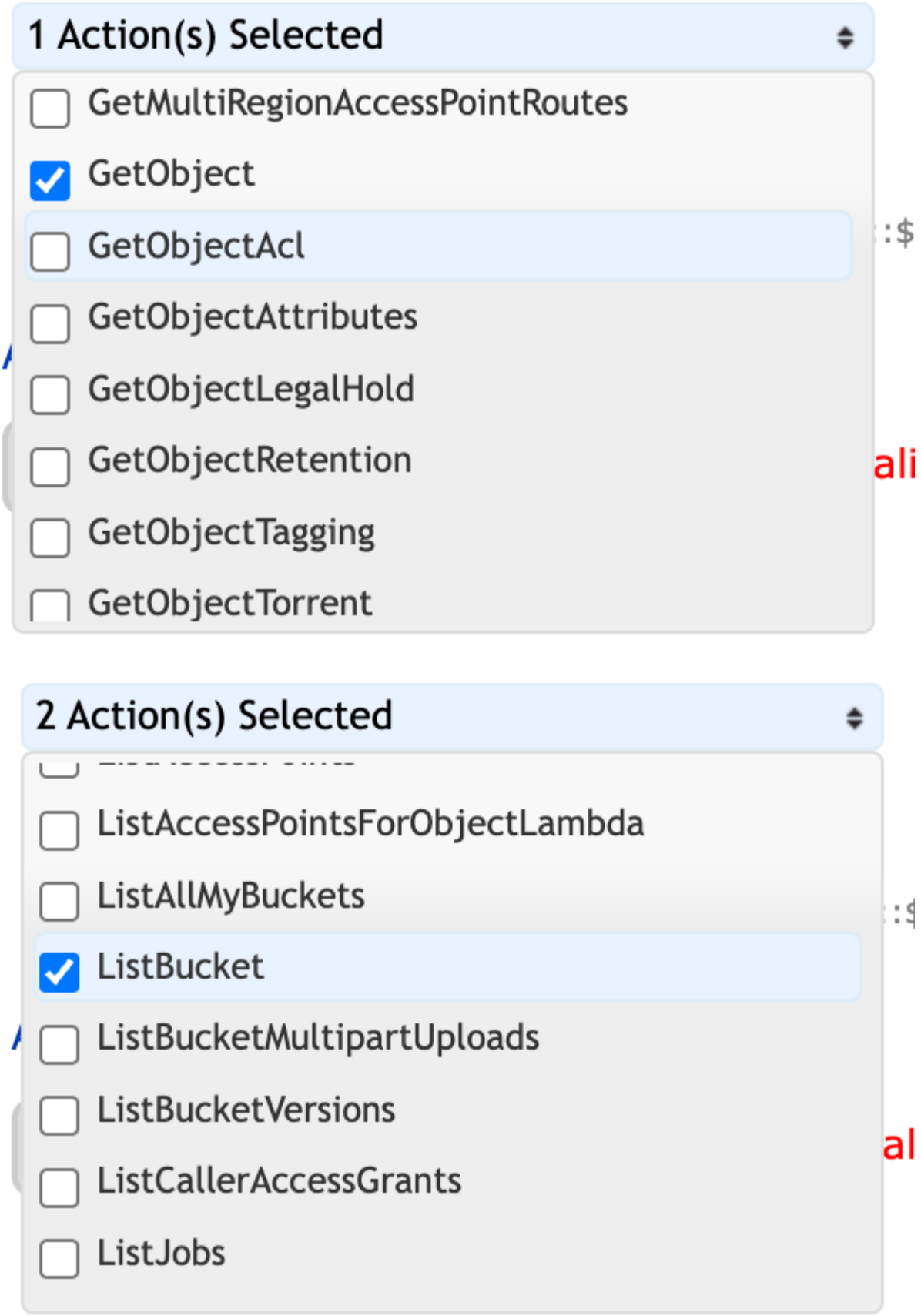
Ad esempio, se si desidera generare un criterio bucket per consentire a tutti gli utenti di eseguire operazioni GetObject su tutti gli oggetti nel bucket, mentre solo gli utenti appartenenti al gruppo "Marketing" nell'account specificato possono accedere completamente.

- Selezionare S3 criterio bucket come tipo di criterio.
- Scegliere l'effetto Consenti
- Inserisci le informazioni del gruppo Marketing - arn:aws:iam::95390887230002558202:Federated-group/Marketing
- Fare clic sulla casella "tutte le azioni"
- Inserisci le informazioni del bucket - arn:aws:S3:::example_bucket,arn:aws:S3:::example_bucket/*

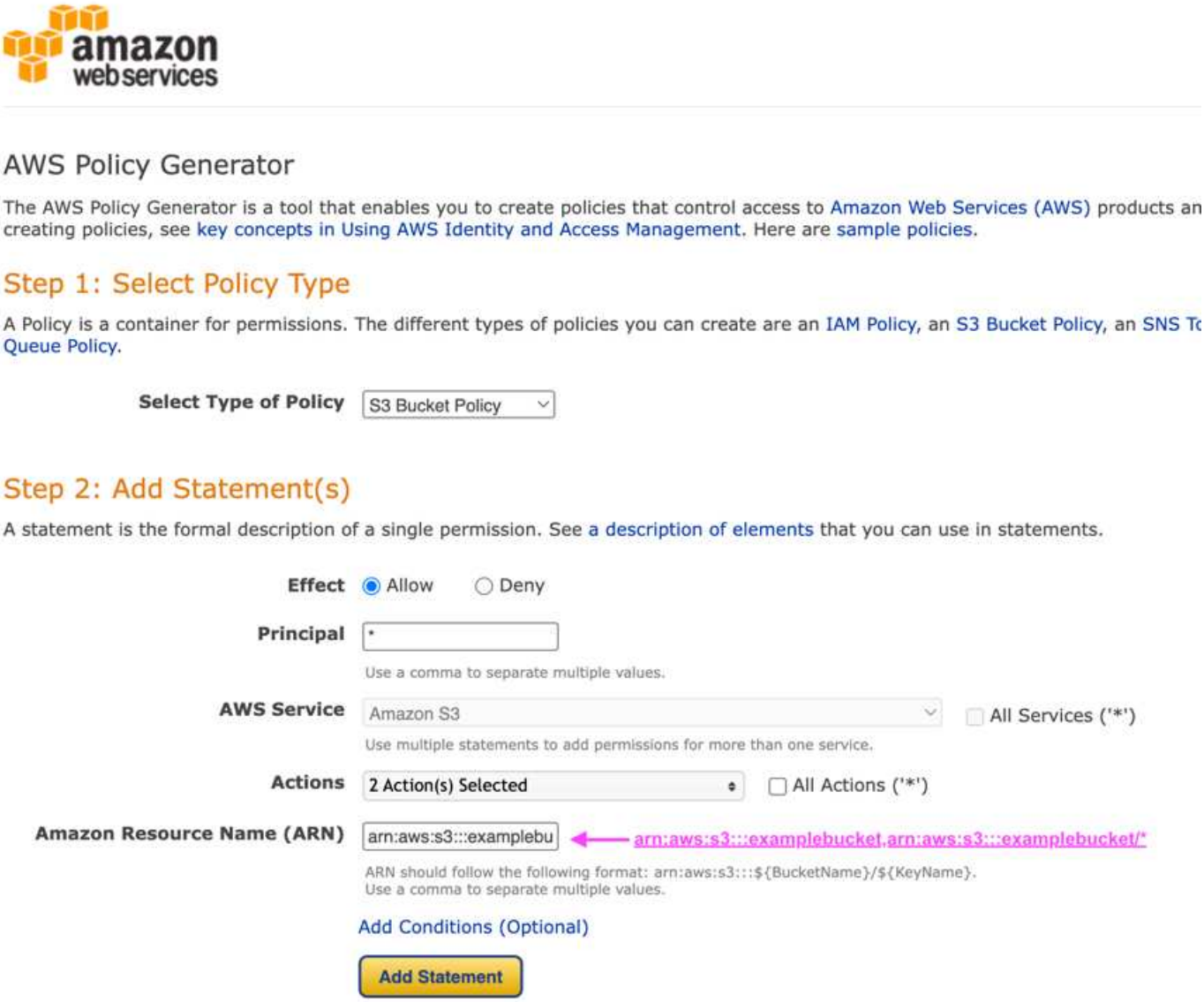


- Fare clic sul pulsante "Aggiungi dichiarazione"

- Scegliere l’effetto Consenti
- Inserisci l’asterisco * per tutti
- Fare clic sulla casella accanto alle azioni GetObject e ListBucket"

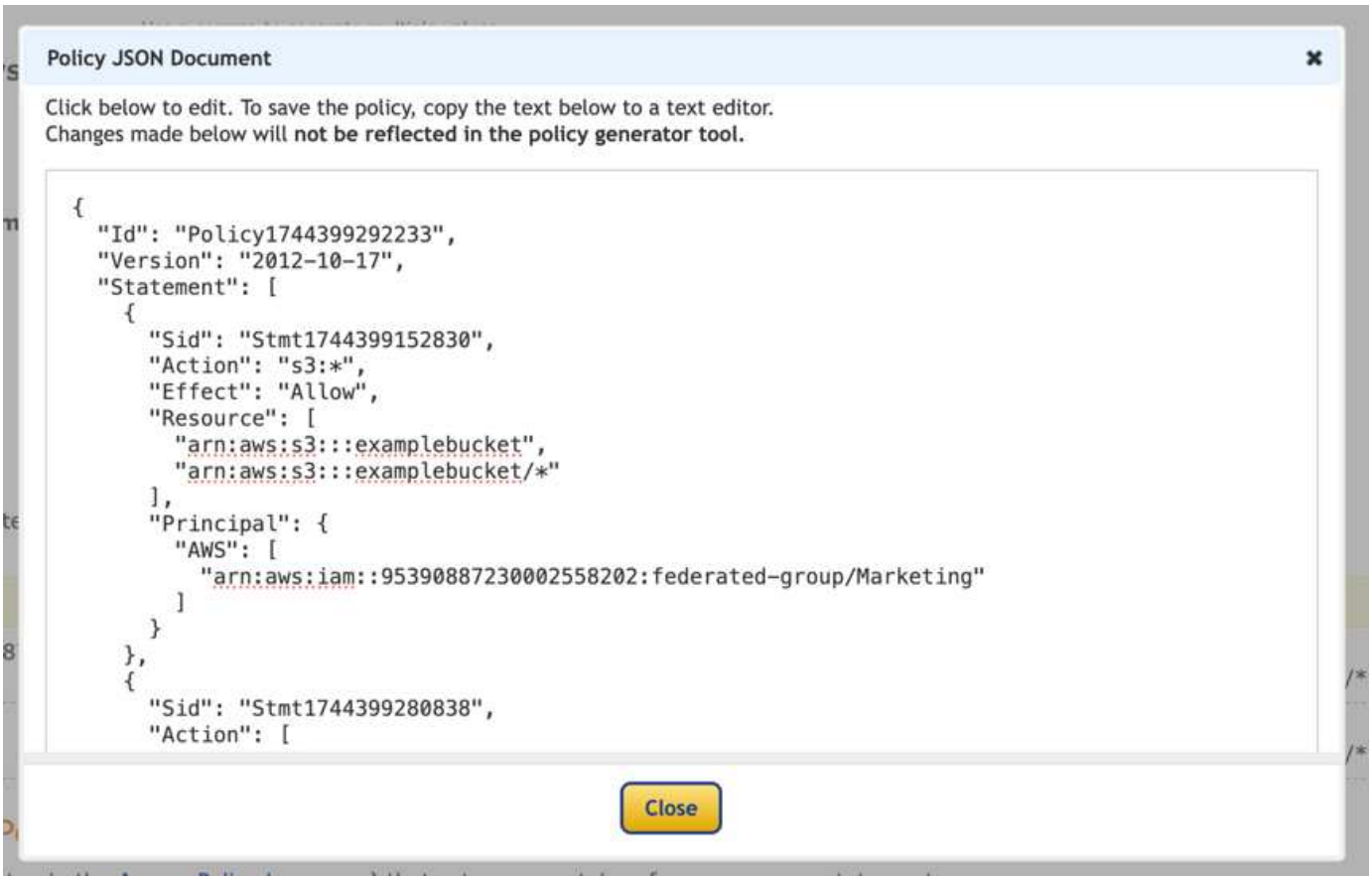


- Inserisci le informazioni del bucket - arn:aws:S3:::example_bucket,arn:aws:S3:::example_bucket/*



- Fare clic sul pulsante "Aggiungi dichiarazione"

- Fare clic sul pulsante "genera criterio" per visualizzare una finestra a comparsa con la policy generata.



- Copiare il testo json completo che dovrebbe avere l'aspetto seguente:

```
{
  "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": "*"
    }
  ]
}
```

Questo json può essere utilizzato così com'è, oppure è possibile rimuovere le righe ID e Version sopra la riga "Statement" e personalizzare il Sid per ogni autorizzazione con un titolo più significativo per ogni autorizzazione o anche questi possono essere rimossi.

Ad esempio:

```
{
  "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": "*"
    }
  ]
}
```

Policy di gruppo (IAM)

Accesso bucket stile home directory

Questo criterio di gruppo consente solo agli utenti di accedere agli oggetti nel bucket denominato 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}/*"
    }

  ]
}
```

Negare la creazione del bucket di blocco degli oggetti

Questo criterio di gruppo limiterà gli utenti a creare un bucket con il blocco degli oggetti attivato nel bucket.

> Questo criterio non viene applicato nell'interfaccia utente di StorageGRID, ma viene applicato solo dall'API S3.

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

Limite di conservazione del blocco degli oggetti

Questa policy di bucket limiterà la durata della conservazione del blocco oggetto a 10 giorni o meno

```
{
 "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"
      }
    }
   }
  ]
}
```

Impedire agli utenti di eliminare gli oggetti in base all'ID versione

Questo criterio di gruppo limita l'eliminazione degli oggetti con versione in base all'ID versione

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

Limitare un gruppo a una singola sottodirectory (prefisso) con accesso in sola lettura

Questo criterio consente ai membri del gruppo di accedere in sola lettura a una sottodirectory (prefisso) all'interno di un bucket. Il nome del bucket è "studio" e la sottodirectory è "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/*"
            ]
        }
    ]
}
```

Criteri benna

Limitare il bucket a un singolo utente con accesso in sola lettura

Questo criterio consente a un singolo utente di avere accesso in sola lettura a un bucket e nega esplicitamente l'accesso a tutti gli altri utenti. Il raggruppamento delle istruzioni Nega in cima alla policy è una buona pratica per una valutazione più rapida.

```
{
    "Statement": [
        {
            "Sid": "Deny non user1",
            "Effect": "Deny",
            "NotPrincipal": {
                "AWS": "arn:aws:iam::349215141330028336 65: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::349215141330028336 65:user/user1"
            },
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::bucket1",
                "arn:aws:s3:::bucket1/*"
            ]
        }
    ]
}
```

limita un bucket a pochi utenti con accesso in sola lettura.

```
{
    "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/*"
        ]
      }
    ]
}
```

Limita l'eliminazione degli oggetti con versione in un bucket da parte dell'utente

Questo criterio bucket limiterà un utente (identificato dall'ID utente "56622399308951294926") a eliminare gli oggetti con versione in base all'ID versione

```
{
  "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"
        ]
      }
    }
  ]
}
```

Ciclo di vita del bucket in StorageGRID

È possibile creare una configurazione del ciclo di vita S3 per controllare quando oggetti specifici vengono cancellati dal sistema StorageGRID.

Che cos'è una configurazione del ciclo di vita

Una configurazione del ciclo di vita è un insieme di regole applicate agli oggetti in specifici bucket S3. Ogni regola specifica quali oggetti sono interessati e quando scadranno (in una data specifica o dopo un certo numero di giorni).

Ciascun oggetto segue le impostazioni di conservazione di un ciclo di vita bucket S3 o di un criterio ILM. Quando viene configurato un ciclo di vita del bucket S3, le azioni di scadenza del ciclo di vita sovrascrivono il criterio ILM per gli oggetti corrispondenti al filtro del ciclo di vita del bucket. Gli oggetti che non corrispondono al filtro del ciclo di vita del bucket utilizzano le impostazioni di conservazione del criterio ILM. Se un oggetto

corrisponde a un filtro del ciclo di vita bucket e non sono specificate esplicitamente azioni di scadenza, le impostazioni di conservazione del criterio ILM non vengono utilizzate ed è implicito che le versioni degli oggetti vengano mantenute per sempre.

Di conseguenza, un oggetto potrebbe essere rimosso dalla griglia anche se le istruzioni di posizionamento in una regola ILM sono ancora applicabili all'oggetto. Oppure, un oggetto potrebbe essere mantenuto sulla griglia anche dopo che tutte le istruzioni di posizionamento ILM per l'oggetto sono scadute

StorageGRID supporta fino a 1,000 regole del ciclo di vita in una configurazione del ciclo di vita. Ogni regola può includere i seguenti elementi XML:

- Scadenza: Consente di eliminare un oggetto quando viene raggiunta una data specificata o quando viene raggiunto un numero di giorni specificato, a partire dalla data di acquisizione dell'oggetto.
- NoncurrentVersionExpiration (NoncurrentExpiration versione): Consente di eliminare un oggetto quando viene raggiunto un numero di giorni specificato, a partire da quando l'oggetto è diventato non corrente.
- Filtro (prefisso, tag)
- Stato *ID

StorageGRID supporta l'utilizzo delle seguenti operazioni bucket per gestire le configurazioni del ciclo di vita:

- DeleteBucketLifecycle
- GetBucketLifecycleConfiguration
- PutBucketLifecycleConfiguration

Struttura di una politica del ciclo di vita

Come primo passo nella creazione di una configurazione del ciclo di vita, è possibile creare un file JSON che includa una o più regole. Ad esempio, questo file JSON include tre regole, come segue:

1. La **Regola 1** si applica solo agli oggetti che corrispondono al prefisso category1/ e che hanno un valore key2 pari a tag2. Il parametro Expiration specifica che gli oggetti che corrispondono al filtro scadranno alla mezzanotte del 22 agosto 2020.
2. La **Regola 2** si applica solo agli oggetti che corrispondono al prefisso category2/. Il parametro Expiration specifica che gli oggetti che corrispondono al filtro scadranno 100 giorni dopo essere stati acquisiti.

   Le regole che specificano un numero di giorni sono relative al momento in cui l'oggetto è stato acquisito. Se la data corrente supera la data di acquisizione più il numero di giorni, alcuni oggetti potrebbero essere rimossi dal bucket non appena viene applicata la configurazione del ciclo di vita.

3. La **Regola 3** si applica solo agli oggetti che corrispondono al prefisso categoria3/. Il parametro Scadenza specifica che tutte le versioni non aggiornate degli oggetti corrispondenti scadranno 50 giorni dopo essere diventate non aggiornate.

```
{
    "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"
        }
    ]
}
```

Applica la configurazione del ciclo di vita al bucket

Dopo aver creato il file di configurazione del ciclo di vita, applicarlo a un bucket inviando una richiesta PutBucketLifecycleConfiguration.

Questa richiesta applica la configurazione del ciclo di vita nel file di esempio agli oggetti in un bucket denominato `testbucket`.

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

Per verificare che una configurazione del ciclo di vita sia stata applicata correttamente al bucket, eseguire una richiesta GetBucketLifecycleConfiguration. Ad esempio:

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

Esempi di policy del ciclo di vita per bucket standard (senza controllo di versione)

Elimina gli oggetti dopo 90 giorni

Caso d'uso: questa policy è ideale per la gestione di dati rilevanti solo per un periodo di tempo limitato, come file temporanei, log o dati di elaborazione intermedi. Vantaggio: riduzione dei costi di archiviazione e garanzia di un bucket ordinato.

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

Esempi di policy del ciclo di vita per bucket con versione

Elimina le versioni non correnti dopo 10 giorni

Caso d'uso: questa policy aiuta a gestire l'archiviazione di oggetti con versioni non aggiornate, che possono accumularsi nel tempo e occupare molto spazio. Vantaggio: ottimizza l'utilizzo dello spazio di archiviazione

mantenendo solo la versione più recente.

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

Mantieni 5 versioni non correnti

Caso d'uso: utile quando si desidera conservare un numero limitato di versioni precedenti a fini di ripristino o controllo. Vantaggio: conservare un numero sufficiente di versioni non correnti per garantire una cronologia e punti di ripristino sufficienti.

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

Rimuovi i marcatori di eliminazione quando non esistono altre versioni

Caso d'uso: questa policy aiuta a gestire i marcatori di eliminazione rimasti dopo la rimozione di tutte le versioni non aggiornate, che possono accumularsi nel tempo. Vantaggio: riduzione di elementi superflui.

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

Elimina le versioni correnti dopo 30 giorni, elimina le versioni non correnti dopo 60 giorni e rimuovi i marcatori di eliminazione creati dall'eliminazione della versione corrente quando non esistono altre versioni.

Caso d'uso: fornire un ciclo di vita completo per le versioni correnti e non correnti, inclusi i marcatori di eliminazione. Vantaggio: ridurre i costi di archiviazione e garantire che il bucket sia ordinato, pur mantenendo un numero sufficiente di punti di ripristino e cronologia.

```
{
  "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
      }
    }
  ]
}
```

rimuovere i marcatori di eliminazione che non hanno altre versioni, conservare 4 versioni non correnti e almeno 30 giorni di cronologia per gli oggetti con il prefisso "accounts_" e conservare 2 versioni e almeno 10 giorni di cronologia per tutte le altre versioni degli oggetti.

Caso d'uso: fornire regole univoche per oggetti specifici e per altri oggetti, per gestire l'intero ciclo di vita delle versioni correnti e non correnti, inclusi i marcatori di eliminazione. Vantaggio: ridurre i costi di archiviazione e garantire che il bucket sia ordinato, pur mantenendo un numero sufficiente di punti di ripristino e cronologia per soddisfare una combinazione di requisiti del cliente.

```
{
  "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
      }
    }
  ]
}
```

Conclusione

- Rivedere e aggiornare regolarmente le policy del ciclo di vita e allinearle agli obiettivi di ILM e di gestione dei dati.
- Testare le policy in un ambiente o in un bucket non di produzione prima di applicarle su larga scala per garantire che funzionino come previsto
- Utilizzare ID descrittivi per le regole per renderle più intuitive, poiché la struttura logica può diventare complessa
- Monitorare l'impatto di queste policy del ciclo di vita dei bucket sull'utilizzo dello storage e sulle prestazioni per apportare le modifiche necessarie.

Informazioni sul copyright

Informazioni sul marchio commerciale