Procedimentos e exemplos de API

StorageGRID solutions and resources

NetApp
November 21, 2025

Índice

Procedimentos e exemplos de API

Teste e demonstre as opções de criptografia S3 no StorageGRID

Por Aron Klein

O StorageGRID e a API S3 oferecem várias maneiras diferentes de criptografar seus dados em repouso. Para saber mais, "Reveja os métodos de encriptação StorageGRID" consulte .

Este guia demonstrará os métodos de criptografia da API S3.

Criptografia do lado do servidor (SSE)

O SSE permite que o cliente armazene um objeto e criptografe-o com uma chave única que é gerenciada pelo StorageGRID. Quando o objeto é solicitado, o objeto é descriptografado pela chave armazenada no StorageGRID.

Exemplo SSE

- COLOQUE um objeto com SSE

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

- DIRIJA o objeto para verificar a criptografia

```
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": {}
}
```

- OBTENHA o objeto

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

Criptografia do lado do servidor com chaves fornecidas pelo cliente (SSE-C)

SSE permite que o cliente armazene um objeto e criptografe-o com uma chave única que é fornecida pelo cliente com o objeto. Quando o objeto é solicitado, a mesma chave deve ser fornecida para descriptografar e retornar o objeto.

Exemplo SSE-C.

- Para fins de teste ou demonstração, você pode criar uma chave de criptografia
  - Crie uma chave de criptografia

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

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

- Coloque um objeto com a chave gerada

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

- Cabeça o objeto

```
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 você não fornecer a chave de criptografia, você receberá um erro "ocorreu um erro (404) ao chamar a operação HeadObject: Not found"

- Obtenha o objeto

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

Se você não fornecer a chave de criptografia, você receberá um erro "ocorreu um erro (InvalidRequest) ao chamar a operação GetObject: O objeto foi armazenado usando uma forma de criptografia do lado do servidor. Os parâmetros corretos devem ser fornecidos para recuperar o objeto."

Criptografia do lado do servidor do bucket (SSE-S3)

O SSE-S3 permite que o cliente defina um comportamento de criptografia padrão para todos os objetos armazenados em um bucket. Os objetos são criptografados com uma chave exclusiva que é gerenciada pelo StorageGRID. Quando o objeto é solicitado, o objeto é descriptografado pela chave armazenada no StorageGRID.

Exemplo SSE-S3 do bucket

- Crie um novo intervalo e defina uma política de criptografia padrão
  - Crie um novo balde

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

  - Coloque criptografia de bucket

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

- Coloque um objeto no balde

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

- Cabeça o objeto

```
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": {}
}
```

- OBTENHA o objeto

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

Teste e demonstre o bloqueio de objetos S3D no StorageGRID

Por Aron Klein

O Object Lock fornece um modelo WORM para impedir que objetos sejam excluídos ou substituídos. A implementação do StorageGRID do bloqueio de objetos é avaliada pela Cohasset para ajudar a atender aos requisitos regulatórios, oferecendo suporte à retenção legal e ao modo de conformidade para retenção de objetos e políticas de retenção de buckets padrão.

Este guia demonstrará a API S3D Object Lock.

Guarda legal

- Bloqueio de objeto retenção legal é um simples status de ligar/desligar aplicado a um objeto.

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

- Verifique-o com uma operação GET.

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

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

- Desligue a retenção legal

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

- Verifique-o com uma operação GET.

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

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

Modo de conformidade

- A retenção de objeto é feita com um carimbo de data/hora retent until.

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

- Verifique o status de retenção

```
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"
    }
}
```

Retenção padrão

- Defina o período de retenção em dias e anos versículos a data de retenção até definida com a api per object.

```
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
```

- Verifique o status de retenção

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

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

- Coloque um objeto no balde

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

- A duração de retenção definida no bucket é convertida em um carimbo de data/hora de retenção no objeto.

```
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"
    }
}
```

Teste a exclusão de um objeto com uma retenção definida

O bloqueio de objetos é construído sobre o controle de versão. A retenção é definida em uma versão do objeto. Se uma tentativa for feita para excluir um objeto com uma retenção definida e nenhuma versão for especificada, um marcador de exclusão será criado como a versão atual do objeto.

- Exclua o objeto com retenção definida

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

- Liste os objetos no intervalo

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

  - Observe que o objeto não está listado.

- Liste versões para ver o marcador de exclusão e a versão original bloqueada

```
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"
        }
    ]
}
```

- Exclua a versão bloqueada do objeto

```
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
```

Políticas e permissões no StorageGRID

Aqui estão exemplos de políticas e permissões no StorageGRID S3.

A estrutura de uma política

No StorageGRID, as políticas de grupo são as mesmas que as políticas de serviço do AWS user (IAM) S3.

As políticas de grupo são necessárias no StorageGRID. Um usuário com S3 chaves de acesso, mas não atribuído a um grupo de usuários, ou atribuído a um grupo sem uma política que conceda algumas permissões, não poderá acessar nenhum dado.

As políticas de bucket e grupo compartilham a maioria dos mesmos elementos. As políticas são construídas no formato json e podem ser geradas usando o. "Gerador de políticas da AWS"

Todas as políticas definirão o efeito, a(s) ação(ões) e o(s) recurso(s). As políticas de bucket também definirão um principal.

O **efeito** será permitir ou negar o pedido.

O **principal**

- Aplica-se apenas a políticas de bucket.
- O principal é a(s) conta(s)/usuário(s) que está sendo concedido(s) ou negado(s) as permissões.
- Pode ser definido como:
  - Um curinga

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

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

  - Um ID de locatário para todos os usuários em um locatário (equivalente à conta da AWS)

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

- Um usuário (local ou federado de dentro do locatário o bucket reside, ou outro locatário na grade)

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

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

- Um grupo (local ou federado de dentro do locatário o bucket reside, ou outro inquilino na grade).

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

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

A **Ação** é o conjunto de S3 operações que estão sendo concedidas ou negadas ao(s) usuário(s).

Para políticas de Grupo, a ação S3:ListBucket permitida é necessária para que os usuários executem quaisquer ações S3D.

O **recurso** é o bucket ou buckets em que os princípios estão sendo concedidos ou negados a capacidade de executar as ações. Opcionalmente, pode haver uma **condição** para quando a ação da política é válida.

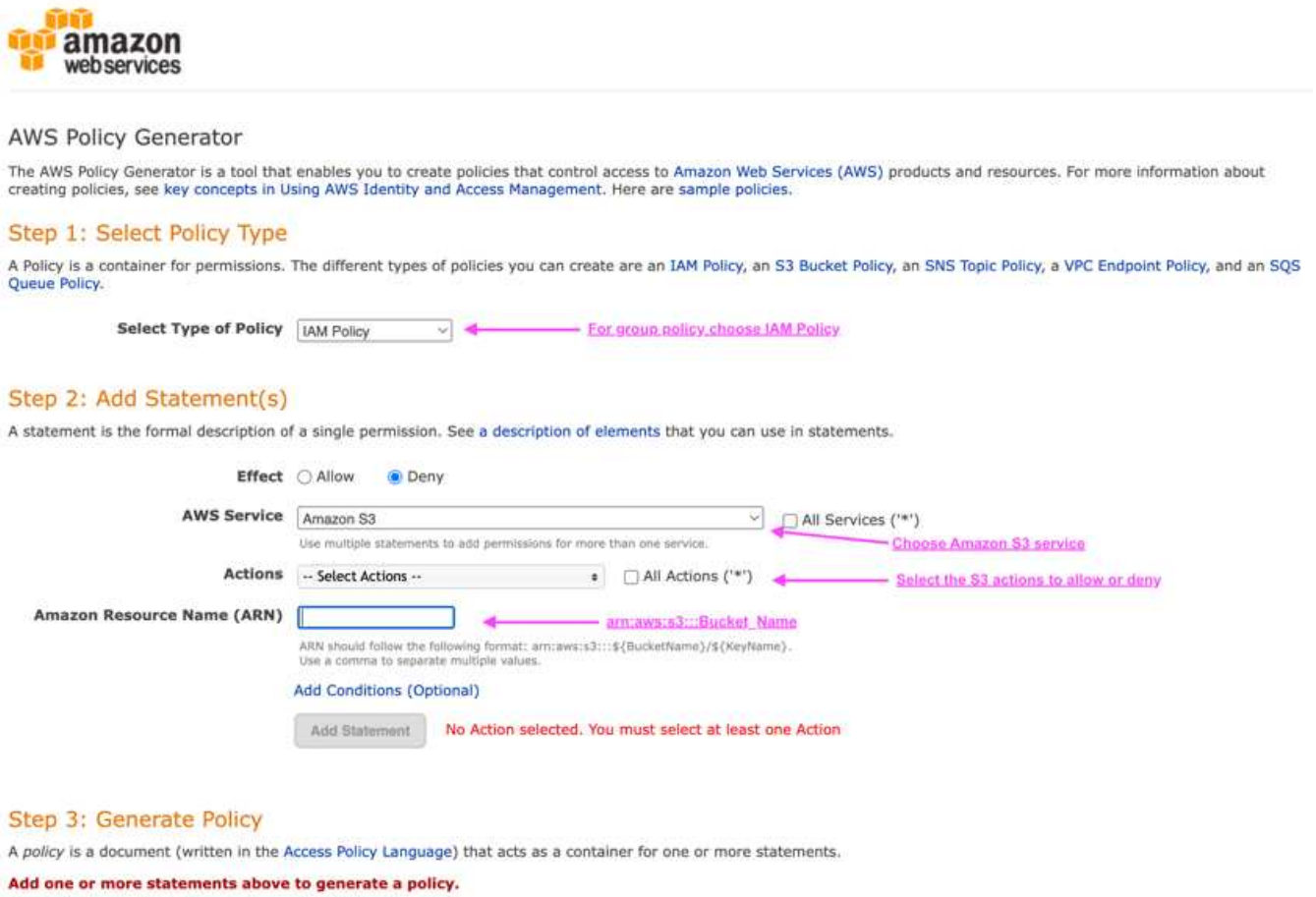
O formato da política JSON será assim:

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

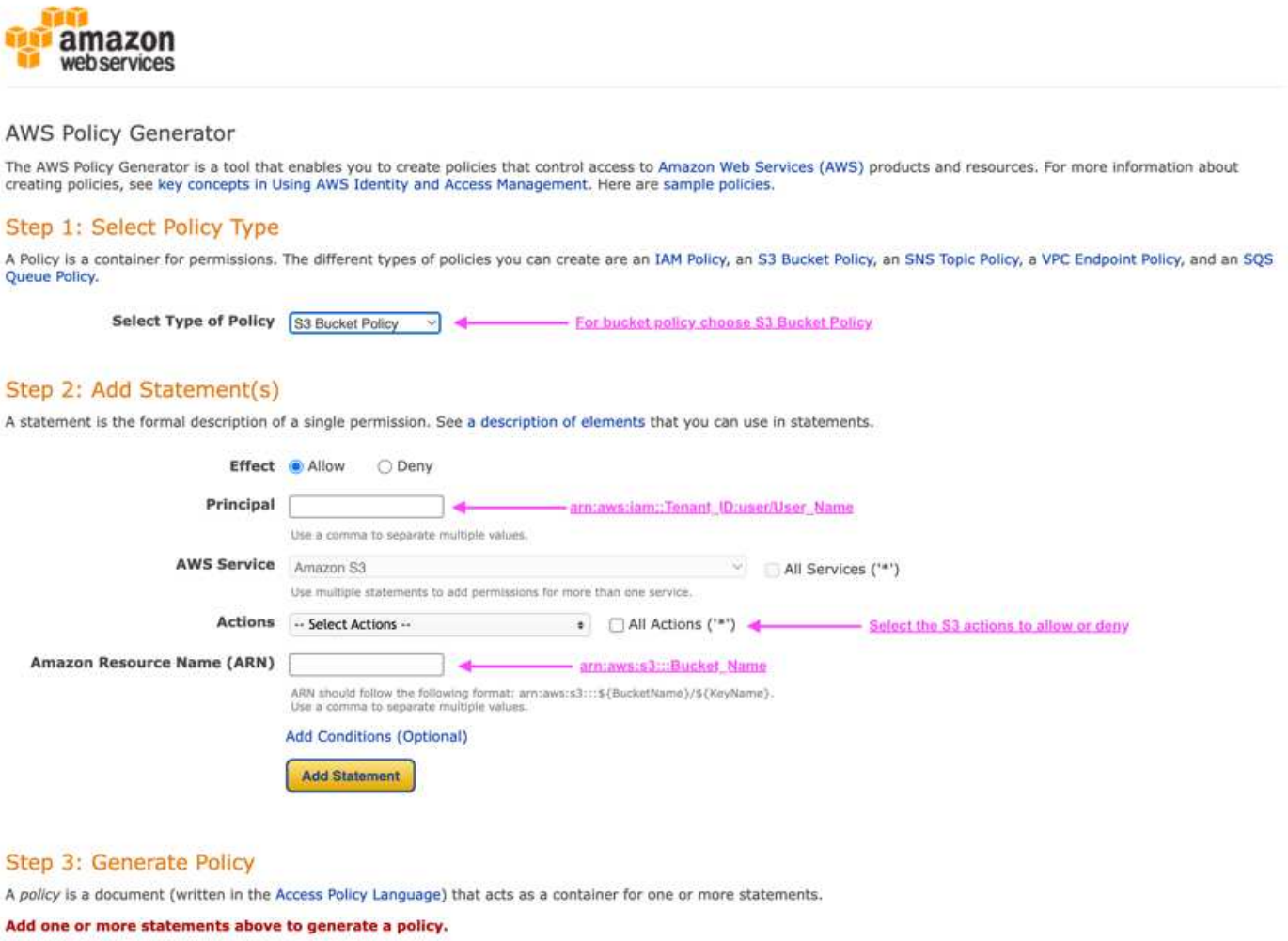
Usando o gerador de políticas da AWS

O gerador de políticas da AWS é uma ótima ferramenta para ajudar a obter o código json com o formato correto e as informações que você está tentando implementar.

Para gerar as permissões para uma política de grupo do StorageGRID: * Escolha a política do IAM para o tipo de política. * Selecione o botão para o efeito desejado - permitir ou negar. É uma boa prática iniciar suas políticas com as permissões de negação e, em seguida, adicionar as permissões de permissão * na caixa suspensa ações clique na caixa ao lado de quantas das ações S3 que você deseja incluir nesta permissão ou na caixa "todas as ações". * Digite os caminhos de intervalo na caixa Nome de recurso do Amazon (ARN). Inclua "ARN:aws:S3:::" antes do nome do intervalo. Ex. "arn:aws:s3:::example_bucket"

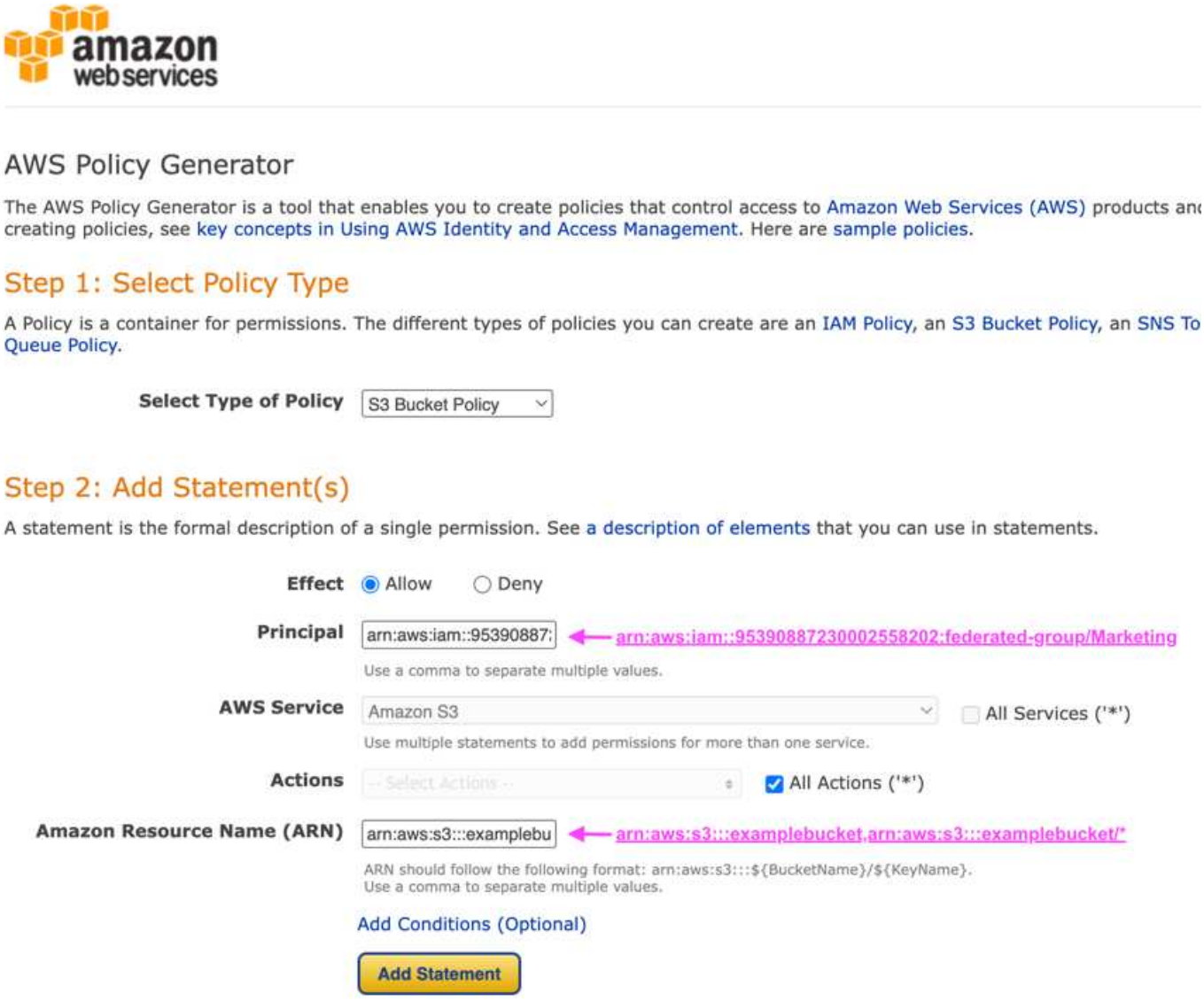


Para gerar as permissões para uma política de bucket: * Escolha a Política de bucket do S3 para o tipo de diretiva. * Selecione o botão para o efeito desejado - permitir ou negar. É uma boa prática iniciar suas políticas com as permissões de negação e, em seguida, adicionar as permissões de permissão * tipo nas informações de usuário ou grupo para o principal. * Na lista suspensa ações, clique na caixa ao lado de tantas das S3 ações que você deseja incluir nesta permissão ou na caixa "todas as ações". * Digite os caminhos de intervalo na caixa Nome de recurso do Amazon (ARN). Inclua "ARN:aws:S3:::" antes do nome do intervalo. Ex. "arn:aws:s3:::example_bucket"



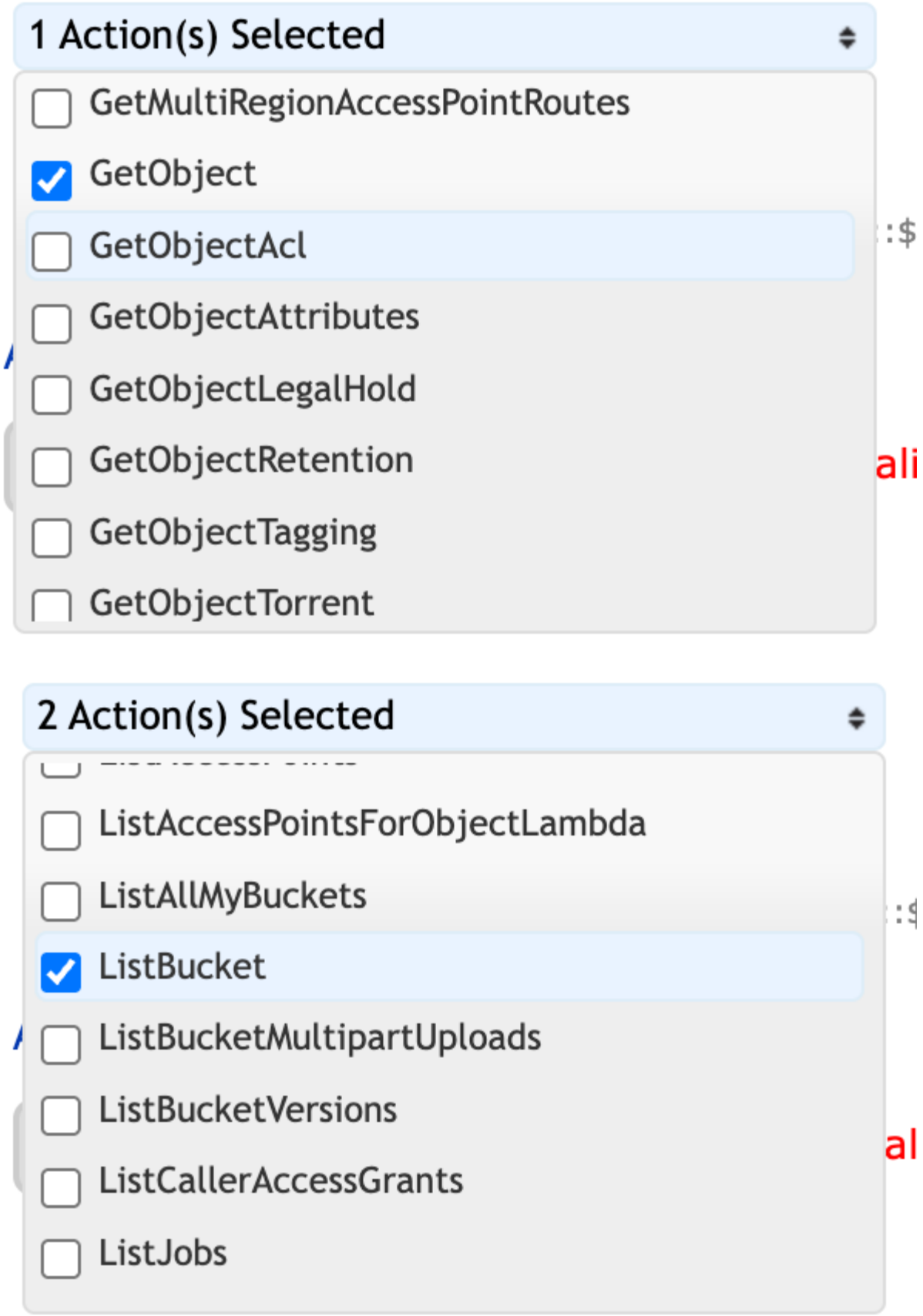
Por exemplo, se você quiser gerar uma política de bucket para permitir que todos os usuários executem operações GetObject em todos os objetos no bucket, enquanto somente os usuários pertencentes ao grupo "Marketing" na conta especificada terão acesso total.

- Selecione S3 Bucket Policy como o tipo de política.
- Escolha o efeito permitir
- Insira as informações do grupo Marketing - ARN:aws:iam::95390887230002558202:grupo federado/Marketing
- Clique na caixa "todas as ações"
- Insira as informações do bucket - ARN:aws:S3:::example_bucket,arn:aws:S3:::example_bucket/*

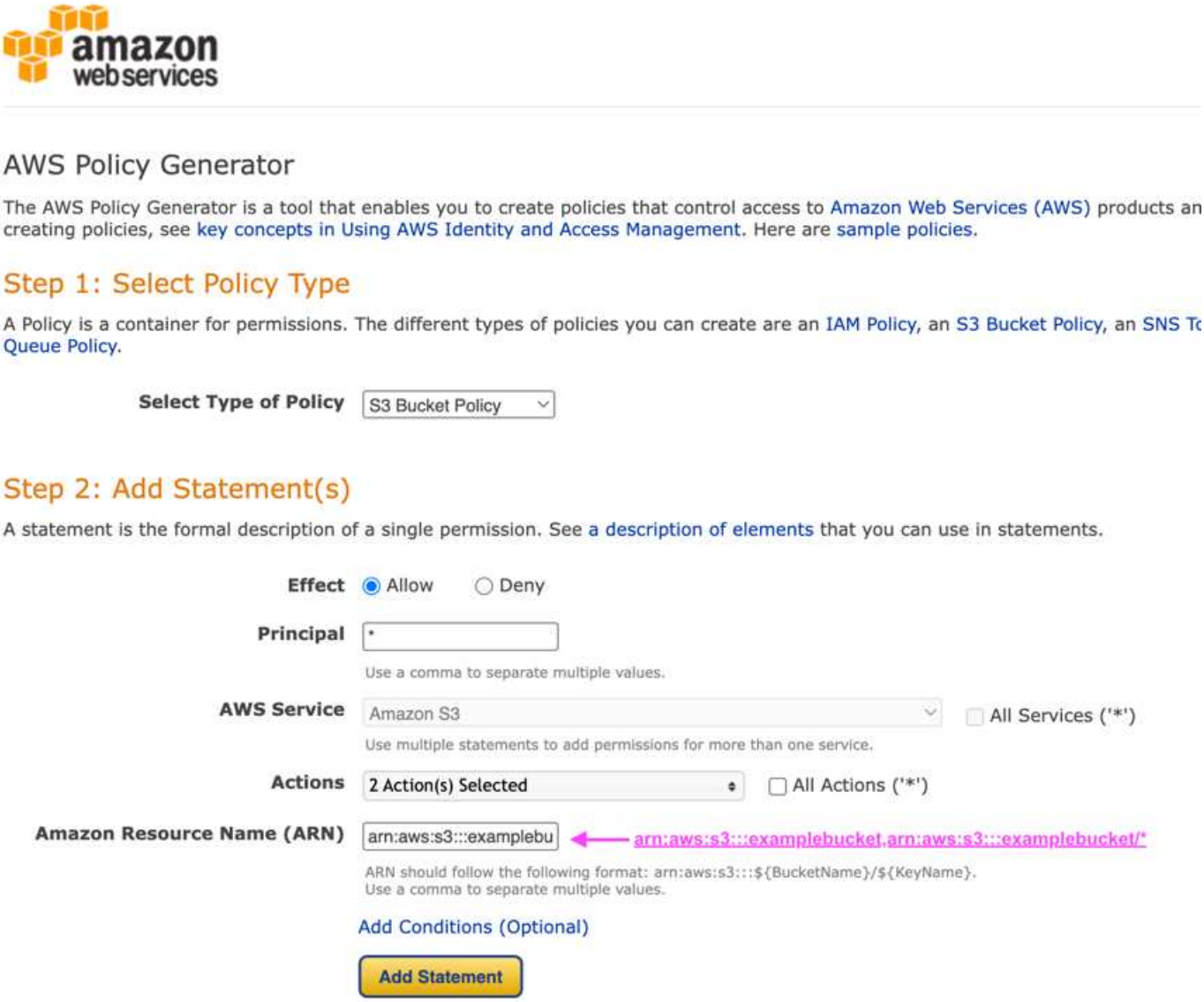


- Clique no botão "Adicionar declaração"

- Escolha o efeito permitir
- Digite o asterisco (*) para todos
- Clique na caixa ao lado de ações GetObject e ListBucket"

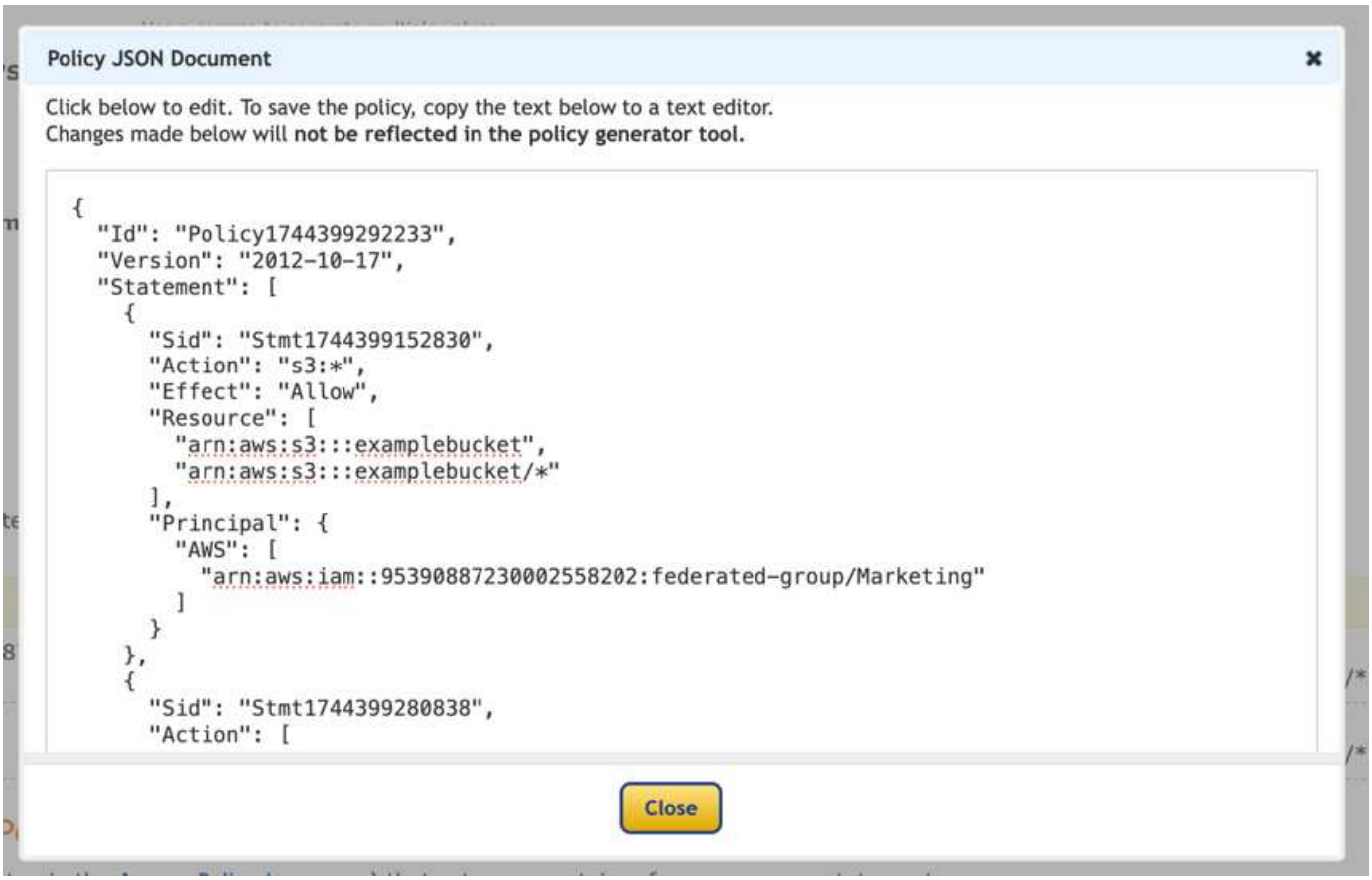


- Insira as informações do bucket - ARN:aws:S3:::example_bucket,arn:aws:S3:::example_bucket/*



- Clique no botão "Adicionar declaração"

- Clique no botão "gerar política" e uma janela pop-up aparecerá com a política gerada.



- Copie o texto json completo que deve ser assim:

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

este json pode ser usado como está, ou você pode remover as linhas ID e versão acima da linha "Statement" e você pode personalizar o Sid para cada permissão com um título mais significativo para cada permissão, ou estes podem ser removidos também.

Por exemplo:

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

Políticas de grupo (IAM)

Acesso ao bucket do estilo do Home Directory

Essa política de grupo só permitirá que os usuários acessem objetos no intervalo chamado nome de usuário do usuário.

```
{
"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}/*"
    }

  ]
}
```

Negar criação de bucket de bloqueio de objetos

Esta política de grupo restringirá os usuários a criar um bucket com o bloqueio de objetos ativado no bucket.

Esta política não é aplicada na IU do StorageGRID, ela só é aplicada pela API S3.

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

Limite de retenção de bloqueio de objetos

Esta política de bucket restringirá a duração de retenção de bloqueio de objetos a 10 dias ou menos

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

Restrinja os usuários de excluir objetos por versionID

Esta política de grupo irá restringir os usuários de excluir objetos versionados por versionID

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

Restrinja um grupo a um subdiretório único (prefixo) com acesso somente leitura

Essa diretiva permite que os membros do grupo tenham acesso somente leitura a um subdiretório (prefixo) dentro de um intervalo. O nome do intervalo é "estudo" e o subdiretório é "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/*"
            ]
        }
    ]
}
```

Políticas do bucket

Restrinja o bucket a um único usuário com acesso somente leitura

Essa política permite que um único usuário tenha acesso somente leitura a um bucket e explicitamente o acesso da denys a todos os outros usuários. Agrupar as declarações deny no topo da política é uma boa prática para uma avaliação mais rápida.

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

restrinja um intervalo a alguns usuários com acesso somente leitura.

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

Restrinja as exclusões do usuário de objetos versionados em um bucket

Esta política de bucket irá restringir um usuário(identificado pelo UserId "56622399308951294926") de excluir objetos versionados por 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"
        ]
      }
    }
  ]
}
```

Ciclo de vida do bucket no StorageGRID

Você pode criar uma configuração de ciclo de vida do S3 para controlar quando objetos específicos são excluídos do sistema StorageGRID.

O que é uma configuração de ciclo de vida

Uma configuração de ciclo de vida é um conjunto de regras que são aplicadas aos objetos em buckets específicos do S3. Cada regra especifica quais objetos são afetados e quando esses objetos expirarão (em uma data específica ou após algum número de dias).

Cada objeto segue as configurações de retenção de um ciclo de vida do bucket do S3 ou de uma política de ILM. Quando um ciclo de vida do bucket do S3 é configurado, as ações de expiração do ciclo de vida substituem a política ILM para objetos que correspondam ao filtro do ciclo de vida do bucket. Os objetos que não correspondem ao filtro do ciclo de vida do bucket usam as configurações de retenção da política ILM. Se

um objeto corresponder a um filtro do ciclo de vida do bucket e nenhuma ação de expiração for explicitamente especificada, as configurações de retenção da política ILM não serão usadas e está implícito que as versões do objeto serão mantidas para sempre.

Como resultado, um objeto pode ser removido da grade, mesmo que as instruções de colocação em uma regra ILM ainda se apliquem ao objeto. Ou um objeto pode ser retido na grade mesmo depois que quaisquer instruções de posicionamento do ILM para o objeto tenham expirado

O StorageGRID dá suporte a até 1.000 regras de ciclo de vida em uma configuração de ciclo de vida. Cada regra pode incluir os seguintes elementos XML:

- Expiração: Exclua um objeto quando uma data especificada é atingida ou quando um número especificado de dias é atingido, a partir de quando o objeto foi ingerido.
- NoncurrentVersionExpiration: Exclua um objeto quando um número especificado de dias é atingido, a partir de quando o objeto se tornou inatual.
- Filtro (prefixo, Tag)
- Status *ID

O StorageGRID dá suporte ao uso das seguintes operações de bucket para gerenciar configurações do ciclo de vida:

- DeleteBucketLifecycle
- GetBucketLifecycleConfiguration
- PutBucketLifecycleConfiguration

Estrutura de uma política de ciclo de vida

Como primeira etapa na criação de uma configuração de ciclo de vida, você cria um arquivo JSON que inclui uma ou mais regras. Por exemplo, este arquivo JSON inclui três regras, como segue:

1. A **Regra 1** aplica-se apenas a objetos que correspondem ao prefixo category1/ e que têm um valor key2 de tag2. O parâmetro Expiration especifica que os objetos que correspondem ao filtro expirarão à meia-noite de 22 de agosto de 2020.
2. A **Regra 2** se aplica apenas a objetos que correspondem ao prefixo category2/. O parâmetro Expiration especifica que os objetos que correspondem ao filtro expirarão 100 dias após serem ingeridos.

   > As regras que especificam um número de dias são relativas a quando o objeto foi ingerido. Se a data atual exceder a data de ingestão mais o número de dias, alguns objetos podem ser removidos do intervalo assim que a configuração do ciclo de vida for aplicada.

3. A **Regra 3** se aplica somente a objetos que correspondem ao prefixo category3/. O parâmetro Expiration especifica que quaisquer versões desatualizadas de objetos correspondentes expirarão 50 dias após se tornarem desatualizadas.

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

Aplique a configuração do ciclo de vida ao bucket

Depois de criar o arquivo de configuração do ciclo de vida, você o aplica a um bucket emitindo uma solicitação PutBucketLifecycleConfiguration.

Essa solicitação aplica a configuração do ciclo de vida no arquivo de exemplo a objetos em um bucket `testbucket` chamado .

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

Para validar que uma configuração de ciclo de vida foi aplicada com sucesso ao bucket, emita uma solicitação GetBucketLifecycleConfiguration. Por exemplo:

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

Exemplo de políticas de ciclo de vida para buckets padrão (sem versão)

Excluir objetos após 90 dias

Caso de uso: Esta política é ideal para gerenciar dados relevantes apenas por um tempo limitado, como arquivos temporários, logs ou dados de processamento intermediário. Benefício: Reduz os custos de armazenamento e garante que o bucket esteja organizado.

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

Exemplo de políticas de ciclo de vida para buckets versionados

Excluir versões não atuais após 10 dias

Caso de uso: Esta política ajuda a gerenciar o armazenamento de objetos de versão desatualizada, que podem se acumular ao longo do tempo e consumir espaço significativo. Benefício: Otimize o uso do

armazenamento mantendo apenas a versão mais recente.

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

Mantenha 5 versões não atuais

Caso de uso: Útil quando você deseja manter um número limitado de versões anteriores para fins de recuperação ou auditoria. Benefício: Manter versões não atuais suficientes para garantir histórico e pontos de recuperação suficientes.

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

Remover marcadores de exclusão quando não houver outras versões

Caso de uso: Esta política ajuda a gerenciar os marcadores de exclusão restantes após a remoção de todas as versões não atuais, que podem se acumular ao longo do tempo. Benefício: Reduz a desordem desnecessária.

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

Exclua as versões atuais após 30 dias, exclua as versões não atuais após 60 dias e remova os marcadores de exclusão criados pela exclusão da versão atual quando não houver mais outras versões.

Caso de uso: Fornecer um ciclo de vida completo para versões atuais e não atuais, incluindo os marcadores de exclusão. Benefício: Reduzir os custos de armazenamento e garantir que o bucket esteja organizado, mantendo pontos de recuperação e histórico suficientes.

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

remova marcadores de exclusão que não tenham outras versões, mantenha 4 versões não atuais e pelo menos 30 dias de histórico para objetos com o prefixo "accounts_" e mantenha 2 versões e pelo menos 10 dias de histórico para todas as outras versões de objetos.

Caso de uso: Forneça regras exclusivas para objetos específicos, juntamente com outros objetos, para gerenciar o ciclo de vida completo das versões atuais e não atuais, incluindo os marcadores de exclusão. Benefício: Reduza os custos de armazenamento e garanta que o bucket esteja organizado, mantendo pontos de recuperação e histórico suficientes para atender a uma variedade de requisitos do 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
      }
    }
  ]
}
```

Conclusão

- Revise e atualize regularmente as políticas de ciclo de vida e alinhe-as com as metas de ILM e gerenciamento de dados.
- Teste as políticas em um ambiente ou bucket não produtivo antes de aplicá-las amplamente para garantir que funcionem conforme o esperado
- Use IDs descritivos para regras para torná-las mais intuitivas, pois a estrutura lógica pode ficar complexa
- Monitore o impacto dessas políticas de ciclo de vida do bucket no uso e no desempenho do armazenamento para fazer os ajustes necessários.

Informações sobre direitos autorais

Informações sobre marcas comerciais