过程和API示例

How to enable StorageGRID in your environment

NetApp
March 07, 2024

目录

过程和API示例

在StorageGRID 上测试和演示S3加密选项

StorageGRID 和S3 API提供了多种不同的方法来加密空闲数据。要了解更多信息，请参见"查看 StorageGRID 加密方法"。

本指南将演示S3 API加密方法。

服务器端加密(SSR)

使用SSE、客户端可以存储对象、并使用由StorageGRID 管理的唯一密钥对其进行加密。请求对象时、该对象将通过存储在StorageGRID 中的密钥进行解密。

SS一 示例

- 使用SSR放置对象

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

- 对对象执行HEAD以验证加密

```
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": {}
}
```

- 获取对象

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

使用客户提供的密钥(SSI-C)进行服务器端加密

通过"SSE "、客户端可以存储对象、并使用客户端随对象提供的唯一密钥对其进行加密。请求对象时、必须提供相同的密钥才能解密并返回对象。

SSI-C示例

- 出于测试或演示目的、您可以创建加密密钥
  - 创建加密密钥

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

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

- 放置具有生成密钥的对象

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

- 为对象设置头

```
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=="
            }
```

如果不提供加密密钥、则会收到错误"An error occurred (404) when calling the HeadObject operation：not found"(调用HeadObject操作时出错(404)：未找到)

- 获取对象

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

如果不提供加密密钥、则在调用GetObject操作时将收到错误"An error occurred (InvalidRequest)：The object was stored using a form of Server side Encryption"。要检索对象、必须提供正确的参数。"

存储分段服务器端加密(SSI-S3)

SSI-S3允许客户端为存储在存储在存储分段中的所有对象定义默认加密行为。这些对象使用由StorageGRID 管理的唯一密钥进行加密。请求对象时、该对象将通过存储在StorageGRID 中的密钥进行解密。

存储分段**SSI-S3**示例

- 创建新存储分段并设置默认加密策略
  - 创建新存储分段

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

  - 放入存储分段加密

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

- 将对象放入存储分段中

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

- 为对象设置头

```
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": {}
}
```

- 获取对象

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

作者：*Aron Klein*

测试并演示StorageGRID 上的S3对象锁定

对象锁定提供了一个WORM模型、用于防止删除或覆盖对象。对StorageGRID 对象锁定实施情况进行了评估、以帮助满足法规要求、支持对象保留的合法保留和合规模式以及默认存储分段保留策略。

本指南将演示S3对象锁定API。

合法保留

- 对象锁定合法保持是应用于对象的简单开/关状态。

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

- 使用GET操作对其进行验证。

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

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

- 关闭合法保留

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

- 使用GET操作对其进行验证。

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

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

合规模式

- 对象保留是使用"保留到"时间戳完成的。

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

- 验证保留状态

```
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"
    }
}
```

默认保留

- 将保留期限设置为天数和年数以及使用每个对象API定义的保留截止日期。

```
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
```

- 验证保留状态

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

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

- 将对象放入存储分段中

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

- 存储分段上设置的保留持续时间将转换为对象上的保留时间戳。

```
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"
    }
}
```

测试删除已定义保留的对象

对象锁定基于版本控制构建。保留是在对象的某个版本上定义的。如果尝试删除定义了保留的对象、但未指定版本、则会创建一个删除标记作为对象的当前版本。

- 删除定义了保留的对象

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

- 列出存储分段中的对象

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

  - 请注意、此对象未列出。

- 列出可查看删除标记的版本以及原始锁定版本

```
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": "566223993089512949 26"
            }
        }
    ],
    "DeleteMarkers": [
        {
            "Owner": {
                "DisplayName": "Tenant01",
                "ID": "566223993089512949 26"
            },
            "Key": "file01.txt",
            "VersionId":
"QjVDQzgzOTAtQ0FGNi0xMUVDLThFMzgtQ0RGMjAwQjk0MjM1",
            "IsLatest": true,
            "LastModified": "2022-05-03T15:35:50.248000+00:00"
        }
    ]
}
```

- 删除对象的锁定版本

```
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
```

作者：*Aron Klein*

分段和组(IAM)策略示例

以下是存储分段策略和组策略(IAM策略)的示例。

组策略(IAM)

主目录模式的存储分段访问

此组策略仅允许用户访问名为Users username的分段中的对象。

```
"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}/*"
    }

  ]
}
```

拒绝创建对象锁定分段

此组策略将限制用户创建在存储分段上启用了对象锁定的存储分段。

> 此策略不会在StorageGRID UI中强制实施、而是仅通过S3 API强制实施。

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

对象锁定保留限制

此存储分段策略会将对象锁定保留期限限制为10天或更短

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

按版本ID限制用户删除对象

此组策略将限制用户按版本ID删除受版本控制的对象

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

此存储分段策略将限制用户(由用户ID "56622399308951294926"标识)按版本ID删除版本控制的对象

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

将存储分段限制为具有只读访问权限的单个用户

此策略允许单个用户对某个存储分段拥有只读访问权限、并明确授予所有其他用户的访问权限。将deny语句分组在策略顶部是一种较好的做法、可以加快评估速度。

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

将组限制为具有只读访问权限的单个子目录(前缀)

此策略允许组成员对分段中的子目录(前缀)具有只读访问权限。分段名称为"study"、子目录为"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/*"
            ]
        }
    ]
}
```