# **ThreatModel for Amazon Simple Storage Service**

### Introduction

Read the blog: [The last S3 security document that we’ll ever need.](https://trustoncloud.com/the-last-s3-security-document-that-well-ever-need/)

### Content

This publication includes:

- overall data flow diagram of Amazon S3

- overview of the Mitre ATT&CK matrix for Amazon S3

- prioritized list of all threat scenarios

- list of all the control activities and testing procedures

- control mappings to PCI DSS [other frameworks available]

- risk-based prioritized list of control implementation

### License Agreement

This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/). This license allows reusers to distribute, remix, adapt, and build upon the material in any medium or format, so long as attribution is given to the creator. The license allows for commercial use. If you remix, adapt, or build upon the material, you must license the modified material under identical terms.



### Source

The latest version of this work is hosted on [GitHub](https://github.com/trustoncloud/threatmodel-for-aws-s3).

### Contact

If you have any questions, please contact [chatbot@trustoncloud.com](mailto:chatbot@trustoncloud.com).

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Amazon Simple Storage Service** Data Flow Diagram | Security Scorecard  |  |  | | --- | --- | | ***Security of the Cloud*** | | | Global Assurance Program coverage (i.e. SOC, PCI, ISO)\* | 6 out of 6 | | Specific Assurance Program coverage (e.g. FedRAMP, HIPAA BAA)\* | 16 out of 16 | |  | | | ***Security in the Cloud*** | | | Number of Actions\* | 186 | | Identity management | AWS IAM, bucket ACL, object ACL | | Number of IAM permissions\* | 157 | | Resource-level statement | Yes | | Resource-based policy | Bucket | | Tag-based ABAC | Yes | | CloudTrail Coverage for APIs | 99.3%  (missing 1) | | Number of CloudTrail Event names\* | 153 | | CloudWatch Events | via CloudTrail | | VPC endpoint | Yes  (Interface + Gateway) | | VPC endpoint policy | Yes | | Network filtering | No | | Encryption-at-rest | Yes  (SSE-KMS, SSE-S3, SSE-C) | | Encryption-in-transit (inc. endpoint protocol) | Yes, but HTTP supported | | CloudFormation | [6](https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/AWS_S3.html) |   \* See details in Appendixes |

## Mitre ATT&CK matrix for Amazon Simple Storage Service

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Initial Access** | **Execution** | **Persistence** | **Privilege Escalation** | **Defense Evasion** | **Credential Access** | **Discovery** | **Lateral Movement** | **Collection** | **Exfiltration** | **Impact** |
| Use a bucket to upload malware or modify an object to include malware [S3.T14] |  | Uncontrolled change in IAM managed policies [S3.T57] | Gain access by modifying or deleting important object tags [S3.T33] | Exfiltrate data via an ungoverned S3 endpoint [S3.T45] |  |  | Reduce bucket security by modify the bucket's Public Access Block [S3.T52] | Move prod data in non-prod environment [S3.T11] | Bucket takeover to gather data [S3.T1] | Grant unauthorized access to a private bucket by changing bucket ACL [S3.T4] |
|  |  |  | Reduce bucket security by deleting the bucket policy [S3.T38] | Evade detection by disabling S3 access logs via bucket ACL change [S3.T51] |  |  | Reduce bucket security by modify the account's Public Access Block [S3.T53] | Unauthorized collection of data by swapping access point [S3.T28] | Unauthorized access to data via bucket replication [S3.T2] | Embed client-side script malware in bucket website [S3.T15] |
|  |  |  |  | Evade detection by disabling S3 access logs via bucket policy change/removal [S3.T58] |  |  | Grant unauthorized access to buckets by changing the Multi-Region Access Point policy [S3.T55] | Use AWS services to access data on S3 [S3.T30] | Exfiltrate your data hosted on an external bucket by using compromised IAM credentials accessed over the Internet [S3.T3] | Files encrypted for ransomware [S3.T16] |
|  |  |  |  | Evade detection by modifying S3 access logs [S3.T59] |  |  |  |  | Unauthorized upload of a private object in an accessible bucket (e.g. public) you do not own [S3.T5] | Destroy or modify primary data [S3.T17] |
|  |  |  |  |  |  |  |  |  | Unauthorized modification of an object to become public or accessible in a private bucket you do not own by changing object ACL [S3.T6] | Hotlinking content from S3 bucket [S3.T22] |
|  |  |  |  |  |  |  |  |  | Exfiltrate data to an attacker bucket via a public endpoint [S3.T7] | Delete objects by using lifecycle [S3.T25] |
|  |  |  |  |  |  |  |  |  | Exfiltrate data by using an S3 VPC endpoint to upload data to an attacker bucket using an internal IAM entity [S3.T8] | Abuse MD5 etag [S3.T27] |
|  |  |  |  |  |  |  |  |  | Exfiltrate data by uploading it to an attacker bucket using a non-authenticated user or an unauthorized external IAM entity via one of your S3 VPC endpoints [S3.T9] | Clickjacking on S3 website [S3.T29] |
|  |  |  |  |  |  |  |  |  | Exfiltrate data by using the public endpoint to upload data in an attacker bucket, using external credentials [S3.T10] | Use of less secure or old S3 features [S3.T35] |
|  |  |  |  |  |  |  |  |  | Intercept data in transit to an external bucket [S3.T12] | Increase bill by creating incomplete uploads [S3.T40] |
|  |  |  |  |  |  |  |  |  | Read data in transit on the website endpoint [S3.T13] | Loss of ownership of an object [S3.T43] |
|  |  |  |  |  |  |  |  |  | Exfiltrate data by using tags [S3.T18] | Exfiltrate, modify or delete objects using Batch [S3.T44] |
|  |  |  |  |  |  |  |  |  | Use CloudFront to access private bucket [S3.T20] | Increase bill by restoring a large amount of data [S3.T47] |
|  |  |  |  |  |  |  |  |  | Exfiltrate data stored on S3 via AWS services [S3.T21] | Affect data protection by removing versioning [S3.T48] |
|  |  |  |  |  |  |  |  |  | Phishing using trademarks [S3.T23] | Affect data protection by removing replication [S3.T49] |
|  |  |  |  |  |  |  |  |  | Unauthorized object restored into an unauthorized bucket [S3.T26] | DoS by blocking traffic using bucket ACL [S3.T50] |
|  |  |  |  |  |  |  |  |  | Upload in an authorized external bucket but an incorrect AWS account [S3.T31] |  |
|  |  |  |  |  |  |  |  |  | Intercept data in transit to an internal bucket [S3.T34] |  |
|  |  |  |  |  |  |  |  |  | Object made public or accessible in a private bucket you own by changing object ACL [S3.T36] |  |
|  |  |  |  |  |  |  |  |  | Grant unauthorized access to a private bucket by changing bucket policy [S3.T37] |  |
|  |  |  |  |  |  |  |  |  | Exfiltrate data by using compromised IAM credentials from the Internet Internet [S3.T39] |  |
|  |  |  |  |  |  |  |  |  | Exfiltrate data via event notification [S3.T41] |  |
|  |  |  |  |  |  |  |  |  | Exfiltrate data via inventory [S3.T42] |  |
|  |  |  |  |  |  |  |  |  | Hijack connection with an Object Lambda [S3.T46] |  |
|  |  |  |  |  |  |  |  |  | Grant unauthorized access to a bucket by changing/deleting access point policy [S3.T54] |  |
|  |  |  |  |  |  |  |  |  | Gain unauthorized access to buckets trusting all Multi-Region Access Points [S3.T56] |  |

## Feature Classes

Amazon Simple Storage Service has the following feature classes and subclasses (i.e. dependent on the usage of its class) that can be activated, restricted, or blocked using AWS Identity and Access Management.

|  |  |  |
| --- | --- | --- |
| **Feature** | **Relation** | **Description** |
| Object operations | class | You can upload, download, and delete virtually any number of objects to an external S3 bucket for which you are authorized. |
| Bucket | subclass of Object operations | To upload your data into your AWS account, you must create an S3 bucket in one of the AWS Regions. |
| Object tagging | subclass of Object operations, used by Bucket | You can tag objects ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/object-tagging.html)). |
| Torrent | subclass of Object operations, used by Bucket | You can use the BitTorrent protocol to retrieve objects ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/S3Torrent.html)). |
| Batch | subclass of Object operations, used by Bucket | S3 Batch Operations performs large-scale Batch Operations on Amazon S3 objects. |
| Object versioning | subclass of Object operations, used by Bucket | You can version your objects ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/ObjectVersioning.html)). |
| Tag on versioned objects | subclass of Object tagging/Object versioning, used by Bucket | You can tag a specific version of an object ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/object-tagging.html)). |
| ACL on versioned objects | subclass of Object versioning/ACL on versioned objects | Amazon S3 Access Control Lists (ACLs) enable you to manage access to object versions ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#permissions)). |
| Bucket versioning | subclass of Object versioning/Bucket | Versioning is a means of keeping multiple variants of an object in the same bucket ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/Versioning.html)). |
| Replication | subclass of Bucket versioning | Replication enables automatic and asynchronous copying of objects of a bucket into another bucket ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/replication.html)). |
| Bucket tag | subclass of Bucket | You can tag buckets ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/CostAllocTagging.html)). |
| Bucket ACL | subclass of Bucket | Amazon S3 Access Control Lists (ACLs) enable you to manage access to buckets ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#permissions)). |
| Bucket policy | subclass of Bucket | For your bucket, you can add a bucket policy to grant other AWS accounts or IAM users permissions for the bucket and the objects in it. Any object permissions apply only to the objects that the bucket owner creates. |
| S3 access logging | subclass of Bucket ACL/Bucket policy | Server access logging provides detailed records for the requests made to a bucket. |
| Analytics | subclass of Bucket | You can analyze storage access patterns to decide the storage class ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/analytics-storage-class.html)). |
| Inventory | subclass of Bucket | You can create a report on your storage, including object metadata or versions ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/storage-inventory.html#storage-inventory-how-to-set-up)). |
| Lifecycle | subclass of Bucket | You can lifecycle your data to reduce the storage cost ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/object-lifecycle-mgmt.html)). |
| Metrics | subclass of Bucket | You can configure metrics to get additional insights into your usage ([ref](https://docs.aws.amazon.com/AmazonS3/latest/user-guide/configure-metrics.html)). |
| S3 Storage Lens | subclass of Bucket | S3 Storage Lens provides a single view of object storage usage and activity across your entire S3 storage. |
| Website | subclass of Bucket | You can host a static website on Amazon S3. On a static website, individual web pages include static content. They might also contain client-side scripts ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/WebsiteHosting.html)). |
| S3 Object Lock | subclass of Bucket | You can use S3 Object Lock to store objects using a write-once-read-many (WORM) model ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/object-lock-overview.html#object-lock-retention-modes)). Creating a bucket with S3 Object Lock will enable versioning even without permissions. |
| Legal hold | subclass of S3 Object Lock | A legal hold provides the same protection as a retention period but has no expiration date. S3 Object Lock must be activated on the bucket. |
| Transfer Acceleration | subclass of Bucket | You can use Transfer Acceleration to improve the performance of long-distance transfers ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/transfer-acceleration.html)). |
| Notification | subclass of Bucket | You can receive notifications when certain events happen in your bucket. |
| Access point | subclass of Bucket | Access points are named network endpoints that are attached to buckets that you can use to perform S3 object operations. |
| S3 Object Lambda | subclass of Access point | S3 Object Lambda enables users to apply their custom code to process the output of a standard S3 request by automatically invoking a Lambda function. |
| Multi-Region Access Points | subclass of Bucket | S3 Multi-Region Access Points provide a single global endpoint to access a data set that spans multiple S3 buckets in different AWS Regions. |
| CORS | subclass of Bucket | You can create a CORS configuration with rules that identify the origins you will allow to access your bucket, the operations (HTTP methods) supported for each origin, and other operation-specific information. |
| Bucket default encryption | subclass of Bucket | You can set default encryption on a bucket so that all new objects are encrypted when stored in the bucket. |
| S3 Object Ownership | subclass of Bucket, used by Object operations | S3 Object Ownership enables bucket owners to automatically assume ownership of objects uploaded to their buckets by other AWS accounts. |
| Public Access Block (bucket) | subclass of Bucket | S3 Block Public Access (bucket) provides controls at the individual S3 bucket level to ensure objects never have public access. |
| Public Access Block (account) | subclass of Bucket | S3 Block Public Access (account) provides controls across an entire AWS account to ensure objects never have public access. |
| Other uses | class | Others can use their S3 service to impact you in some ways. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Object operations *(class, FC1)* *You can upload, download, and delete virtually any number of objects to an external S3 bucket for which you are authorized. Amazon S3 Access Control Lists (ACLs) enable you to manage access to objects. Each object has an ACL attached to it as a sub-resource. It defines which AWS accounts or groups are granted access and the type of access. When a request is received against a resource, Amazon S3 checks the corresponding ACL to ensure the requester has the necessary access permissions (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#permissions)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Retrieves an object from Amazon S3. | s3:GetObject | | Adds an object to a bucket. | s3:PutObject | | Sets the Access Control List (ACL) permissions for an object. You must have WRITE\_ACP permission to set the ACL of an object. | s3:PutObjectAcl |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Loss of ownership of an object | [High (7.1)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:N) | | Exfiltrate data by uploading it to an attacker bucket using a non-authenticated user or an unauthorized external IAM entity via one of your S3 VPC endpoints | [Medium (6.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N) | | Destroy or modify primary data | [Medium (6.1)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:N/I:H/A:H) | | Exfiltrate data stored on S3 via AWS services | [Medium (5.8)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:L/UI:N/S:C/C:H/I:N/A:N) | | Exfiltrate data to an attacker bucket via a public endpoint | [Medium (5.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | Exfiltrate your data hosted on an external bucket by using compromised IAM credentials accessed over the Internet | [Medium (5.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | Unauthorized upload of a private object in an accessible bucket (e.g. public) you do not own | [Medium (5.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | Exfiltrate data by using an S3 VPC endpoint to upload data to an attacker bucket using an internal IAM entity | [Medium (5.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | Unauthorized modification of an object to become public or accessible in a private bucket you do not own by changing object ACL | [Medium (5.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:L/I:H/A:N) | | Intercept data in transit to an external bucket | [Medium (4.6)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:P/AC:H/PR:N/UI:R/S:U/C:H/I:L/A:N) | | Unauthorized object restored into an unauthorized bucket | [Medium (4.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:R/S:U/C:H/I:N/A:N) | | Upload in an authorized external bucket but an incorrect AWS account | [Medium (4.0)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:H/UI:R/S:U/C:H/I:N/A:N) | | Exfiltrate data via an ungoverned S3 endpoint | [Low (1.9)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:U/C:N/I:L/A:N) | | Use of less secure or old S3 features | [Low (1.9)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:U/C:N/I:L/A:N) | |

#### Loss of ownership of an object

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T43 | | **Name** | Loss of ownership of an object | | **Description** | S3 Object Ownership enables a bucket receiver to convert a bucket-owner-full-control ACL into an ownership transfer (for a new object); additionally, a bucket can convert all the objects to be owned by the bucket owner. An attacker can modify the receiver bucket to remove your object ACL control on an object and remove your access to this object. | | **Goal** | Data manipulation | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [High (7.1)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:N) | | **IAM Access** | {  "OPTIONAL": "s3:PutBucketOwnershipControls"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Disabling ACLs for all buckets**    Ensure bucket ACL and object ACL are disabled on each bucket    Prevent the creation of buckets with ACL enabled (e.g. by using a SCP and/or an IAM policy on "s3:CreateBucket" with a deny statement on StringNotEquals "s3:x-amz-object-ownership":"BucketOwnerEnforced"). Note that it does not block someone from enabling an ACL afterward via PutPutBucketOwnershipControls. | Very High | 1 | 1 | - |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    For all external buckets with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover, block the PutObject with any ACL (e.g. using IAM or SCP and a deny on the condition "StringLike": {"s3:x-amz-acl": "\*"}). It should be called via PutObjectAcl.    For all external bucket with bucket-owner-full-control ACL requirements but without S3 Object Ownership handover, monitor that the PutObject do not include the ACL operation | Very High | 1 | 1 | 1 |
| **Enforce good coding practice**    When transmitting an object to an external bucket with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover, use 2 separate APIs (PutObject and PutObjectAcl) instead of the built-in object ACL operation in PutObject. | High | 1 | - | - |

#### Exfiltrate data by uploading it to an attacker bucket using a non-authenticated user or an unauthorized external IAM entity via one of your S3 VPC endpoints

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T9 | | **Name** | Exfiltrate data by uploading it to an attacker bucket using a non-authenticated user or an unauthorized external IAM entity via one of your S3 VPC endpoints | | **Description** | VPC endpoints for S3 allow any entity to connect from a VPC to any S3 bucket without an Internet Gateway. An attacker can exfiltrate data to an external S3 bucket via one of your VPC endpoints, using a non-authenticated user or its own external IAM entity. Note that some external IAM entities might be authorized if provided by one of your business partners. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (6.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Restrict access point access to VPC when in use**    Maintain a list of authorized access between VPCs, S3 access points, and S3.    Limit access via the S3 access point by using a VPC endpoint and/or bucket policy with the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" in an allow statement to reduce the length of the allowlist bucket name in the VPC endpoint/bucket policy. | Very High | 1 | 1 | - |
| **Limit and monitor access via S3 VPC endpoints**    For each VPC, maintain a list of AWS Organizations, OU, and/or AWS account(s) where IAM entities are authorized to access S3    For each VPC with an IAM entity allowed to use S3, secure them with the VPC ThreatModel (e.g. [modification of VPC endpoints, VPC endpoint policy, routing table, Security Groups](https://docs.aws.amazon.com/vpc/latest/userguide/vpce-gateway.html))    Block any IAM entity not belonging to an authorized AWS Organizations, OU, and/or AWS account(s) to call S3 from your VPCs by adding a deny statement on the S3 VPC endpoint policy of each VPC, with the condition using "aws:PrincipalOrgPaths" ([ref](https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_condition-keys.html#condition-keys-principalorgpaths)) including the full Org ID, as those are globally unique.    Enable [VPC DNS query logging](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resolver-query-logs.html) in all VPC    Maintain a list of authorized S3 and S3 access point (and their respective AWS accounts) to be accessed for each VPC    Limit the access to only authorized S3 bucket(s) or their AWS account(s) from each VPC (e.g. using the condition key "s3:ResourceAccount" on the VPC endpoint policy, alternatively use a specific resource-level statement for each bucket, or if the VPC endpoint policy size is beyond the limit and more granular control on VPC is required, use access points)    Monitor VPC DNS query logs that only authorized S3 bucket and S3 access points are being queried in each VPC (e.g. using VPC DNS query logging), and protect it using Route53 ThreatModel | Very High | 4 | 2 | 1 |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    Request access via an S3 access point on a bucket you don't own, if compatible with your interaction with the bucket (e.g. not through not-compatible AWS service) | High | 2 | - | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |

#### Destroy or modify primary data

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T17 | | **Name** | Destroy or modify primary data | | **Description** | S3 provides high durability by design (11 9s). However, data can still be deleted by the customer. An attacker (or someone by negligence) can use its access to destroy (or modify) primary data located on S3, affecting the ability of the business to operate (for example, [Code Spaces](https://www.networkcomputing.com/cloud-infrastructure/code-spaces-lesson-cloud-backup)). | | **Goal** | Disruption of Service | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Medium (6.1)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:N/I:H/A:H) | | **IAM Access** | {  "AND": [{  "OR": ["s3:DeleteObject", "s3:PutObject"]  }, {  "OPTIONAL": "s3:DeleteObjectVersion"  }, {  "OPTIONAL": "s3:BypassGovernanceMode"  }]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Identify and ensure the protection of all internal buckets hosting your objects**    Track all buckets you control, define their authorized data classification, identify whether the hosted data is primary (i.e. source of truth, for example logs, backups, forensic data, raw data, etc.) or an input/output of a process (e.g. file-processing, software package, etc.), their WORM requirements (e.g. SEC 17a-4, CTCC, etc.), if they are production/non-production (preferably done at account-level), their storage class. You may use tags, Infra-as-code, AWS Glue Data Catalog, or external management tools like [FINRA herd](https://finraos.github.io/herd/)) | Very High | 1 | - | - |
| **Enforce encryption-at-rest**    Maintain a list of authorized KMS key(s) for each bucket and their default encryption key. You might simplify by using only 1 key per bucket, ideally dedicated. Note that an S3 server access log bucket does not support KMS encryption ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/bucket-encryption.html#bucket-encryption-how-to-set-up)).    Ensure all objects on S3 buckets are encrypted with an authorized KMS key    Use KMS ThreatModel to protect the KMS keys used for S3 (e.g. using encryptionContext on the policy of each KMS key)    Implement an authorized default encryption key on each bucket; and enable S3 Bucket Key, if CloudTrail events are not required for KMS encrypt/decrypt (note: Amazon S3 evaluates and applies bucket policies before applying bucket default encryption settings)    Block PutObject requests with unauthorized KMS key on each bucket (e.g. using an S3 bucket policy deny statement on PutObject if the condition if exists "s3:x-amz-server-side-encryption-aws-kms-key-id" is not an authorized KMS key) | Very High | 4 | 1 | - |
| **Protect primary data against loss**    Enable versioning on buckets holding primary data    Backup primary data in a secure location under a different security authority (e.g. in an [AWS data bunker account](https://wellarchitectedlabs.com/security/100_labs/100_create_a_data_bunker/1_instructions/) via replication, or using AWS Backup for Amazon S3) | Very High | 2 | - | - |
| **Use S3 Object Lock to protect data integrity**    Implement the authorized default S3 Object Lock on each bucket (note: Amazon S3 evaluates and applies bucket policies before applying bucket default S3 Object Lock settings)    Block PutObject and PutObjectRetention requests with unauthorized S3 Object Lock on each bucket (e.g. using an S3 bucket policy deny statement on PutObject and PutObjectRetention if the condition if exists "s3:object-lock-mode" and "s3:object-lock-remaining-retention-days" is not the defined S3 Object Lock configuration) | Very High | - | 2 | - |
| **Limit access from only authorized VPCs**    For each S3 bucket, maintain a list of VPC(s) authorized to access it.    Limit the access to only those VPC(s) (e.g. using S3 bucket statement, deny if the condition "aws:SourceVpce", or if the bucket policy size is beyond the limit, use this condition on access point) | High | 1 | 1 | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | High | 2 | - | - |
| **Enforce good coding practice**    Use "x-amz-checksum" from the [object metadata](https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingMetadata.html#SysMetadata) to validate the integrity of the object instead of etag. If etag is used, make sure properly account for its different definitions ([ref](https://teppen.io/2018/06/23/aws_s3_etags/)) | Medium | 1 | - | - |
| **Have a process to apply legal hold**    Create a process to apply a legal hold to any S3 bucket whenever required. The condition "s3:object-lock-legal-hold" can be used to restrict who can remove such a lock. | Medium | 1 | - | - |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Low | 1 | - | - |

#### Exfiltrate data stored on S3 via AWS services

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T21 | | **Name** | Exfiltrate data stored on S3 via AWS services | | **Description** | Number of AWS services are using S3 for storage, including storing in cross-account S3 buckets. Services with IAM roles (e.g. SageMaker) will give ownership to the target AWS account, removing ownership protection. An attacker can use those services to exfiltrate data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.8)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:L/UI:N/S:C/C:H/I:N/A:N) | | **IAM Access** | {  "OPTIONAL": "s3:PutObjectAcl"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    Monitor that only authorized external buckets are used (e.g. via CloudTrail S3 data events in resources[].accountId and resources[].ARN). Both account ID and bucket name must be verified. | High | 1 | - | 1 |
| **Model the threats on all AWS services accessing S3**    Analyze and protect all AWS services accessing S3 (e.g. via ThreatModel). Enforce usage in VPC only, whenever possible. | High | 1 | - | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Low | 2 | - | - |

#### Exfiltrate data to an attacker bucket via a public endpoint

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T7 | | **Name** | Exfiltrate data to an attacker bucket via a public endpoint | | **Description** | S3 allows IAM entities to upload data in a bucket in other AWS accounts, if they have the IAM permissions. An attacker can use one of your IAM entities to upload data to one of their buckets. If the attacker does not control object ACL, it can use the name of objects (1KB). | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "AND": ["s3:PutObject", {  "OPTIONAL": "s3:PutObjectAcl"  }]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block S3 endpoints in your corporate perimeter security**    Block S3 endpoints ([DNS](https://docs.aws.amazon.com/general/latest/gr/s3.html) and [IP ranges](https://aws.amazon.com/premiumsupport/knowledge-center/s3-find-ip-address-ranges/)) in your corporate perimeter security to the Internet (e.g. firewalls, or cloud interception proxy like [Kivera](https://kivera.io)) including via Internet Gateway, to force usage of VPC endpoints. It will block data-plane transfer. Note: AWS console stays functional as it proxies non-data-plane requests (via "console.aws.amazon.com"). | Very High | 1 | - | - |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    Monitor that only authorized external buckets are used (e.g. via CloudTrail S3 data events in resources[].accountId and resources[].ARN). Both account ID and bucket name must be verified. | High | 1 | - | 1 |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Restrict access point access to VPC when in use**    Maintain a list of authorized access between VPCs, S3 access points, and S3.    Limit access via the S3 access point by using a VPC endpoint and/or bucket policy with the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" in an allow statement to reduce the length of the allowlist bucket name in the VPC endpoint/bucket policy.    In the S3 bucket policy, deny all IAM principals not using an authorized S3 access point(s) using the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn"    Block the creation "s3:CreateAccessPoint" of non-VPC S3 access point (e.g. using the condition "StringNotEquals": {"s3:AccessPointNetworkOrigin": "VPC"})    Block all traffic from Internet-configured S3 access point (e.g. on the bucket policy, using a deny statement with the condition "StringNotEquals": {"s3:AccessPointNetworkOrigin": "VPC"})    Block any [object-related operations](https://docs.aws.amazon.com/AmazonS3/latest/dev/using-access-points.html#access-points-service-api-support) access to S3 bucket not through access point (i.e. using a deny IAM policy statement with the condition "ArnNotLike" {"s3:DataAccessPointArn": "arn:aws:s3:*Region*:*AccountId*:accesspoint/\*"}) | Medium | 1 | 5 | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Low | 1 | - | - |

#### Exfiltrate your data hosted on an external bucket by using compromised IAM credentials accessed over the Internet

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T3 | | **Name** | Exfiltrate your data hosted on an external bucket by using compromised IAM credentials accessed over the Internet | | **Description** | IAM credentials can be compromised. An attacker can use a compromised but authorized credential to download your object from an external bucket via the public endpoint or their VPC endpoint. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:GetObject"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel). | High | 1 | - | - |
| **Enforce good coding practice**    Ensure all S3 buckets interacted with are in the correct AWS account (e.g. using the condition in all compatible S3 requests: x-amz-expected-bucket-owner and x-amz-source-expected-bucket-owner) | Medium | 1 | - | - |
| **Monitor S3 with Amazon GuardDuty and Macie**    Enable and monitor [S3 protection in Amazon GuardDuty](https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_finding-types-s3.html) in all AWS accounts in all Regions, and protect it using GuardDuty ThreatModel. Ensure findings are investigated (e.g. using Amazon Detective). | Low | 1 | - | - |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Very Low | 1 | - | - |

#### Unauthorized upload of a private object in an accessible bucket (e.g. public) you do not own

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T5 | | **Name** | Unauthorized upload of a private object in an accessible bucket (e.g. public) you do not own | | **Description** | S3 buckets can be public for a legitimate reason. An attacker (or someone by negligence) can upload sensitive data in an accessible bucket (e.g. public) you do not own to make it accessible to exfiltrate it. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutObject"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    Allow only authorized ACL on objects for buckets you don't control (e.g. using IAM and VPC endpoint policy with the [ACL conditions](https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#acl-specific-condition-keys))    Scan all data before uploading to an external bucket to ensure the classification of the data is aligned with the bucket classification (e.g. using Macie). | High | 2 | 1 | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Low | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Very Low | 1 | - | - |

#### Exfiltrate data by using an S3 VPC endpoint to upload data to an attacker bucket using an internal IAM entity

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T8 | | **Name** | Exfiltrate data by using an S3 VPC endpoint to upload data to an attacker bucket using an internal IAM entity | | **Description** | VPC endpoints for S3 allow IAM entities to connect from a VPC to any S3 bucket without an Internet Gateway. An attacker can exfiltrate pre-collected data to an external S3 bucket via a VPC endpoint, using an internal IAM entity they control. If the attacker does not control object ACL, it can use the name of objects (1KB). | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "AND": ["s3:PutObject", {  "OPTIONAL": "s3:PutObjectAcl"  }]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    Request access via an S3 access point on a bucket you don't own, if compatible with your interaction with the bucket (e.g. not through not-compatible AWS service) | High | 2 | - | - |
| **Limit and monitor access via S3 VPC endpoints**    Enable [VPC DNS query logging](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resolver-query-logs.html) in all VPC    Maintain a list of authorized S3 and S3 access point (and their respective AWS accounts) to be accessed for each VPC    Limit the access to only authorized S3 bucket(s) or their AWS account(s) from each VPC (e.g. using the condition key "s3:ResourceAccount" on the VPC endpoint policy, alternatively use a specific resource-level statement for each bucket, or if the VPC endpoint policy size is beyond the limit and more granular control on VPC is required, use access points)    Monitor VPC DNS query logs that only authorized S3 bucket and S3 access points are being queried in each VPC (e.g. using VPC DNS query logging), and protect it using Route53 ThreatModel | High | 2 | 1 | 1 |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

#### Unauthorized modification of an object to become public or accessible in a private bucket you do not own by changing object ACL

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T6 | | **Name** | Unauthorized modification of an object to become public or accessible in a private bucket you do not own by changing object ACL | | **Description** | Bucket authority only prevails on object ACL when the object access is explicitly denied ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/access-control-auth-workflow-object-operation.html)). An attacker (or someone by negligence) can change the object ACL to make it public or accessible for themselves. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:L/I:H/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutObjectAcl"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block changes to make an object public via object ACL**    Deny requests to change object ACL to public (e.g. using an SCP, S3 bucket policy, and VPC endpoint policy blocking PutObjectAcl for "s3:x-amz-grant-read", "s3:x-amz-grant-read-acp", "s3:x-amz-grant-write-acp", "s3:x-amz-grant-full-control" on the following predefined groups "http://acs.amazonaws.com/groups/global/AllUsers" and "http://acs.amazonaws.com/groups/global/AuthenticatedUsers")    Monitor ObjectACL changed (or tentatively changed) to public using CloudTrail S3 data events | Very High | - | 1 | 1 |
| **Disabling ACLs for all buckets**    Ensure bucket ACL and object ACL are disabled on each bucket    Prevent the creation of buckets with ACL enabled (e.g. by using a SCP and/or an IAM policy on "s3:CreateBucket" with a deny statement on StringNotEquals "s3:x-amz-object-ownership":"BucketOwnerEnforced"). Note that it does not block someone from enabling an ACL afterward via PutPutBucketOwnershipControls. | Very High | 1 | 1 | - |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    Allow only authorized ACL on objects for buckets you don't control (e.g. using IAM and VPC endpoint policy with the [ACL conditions](https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#acl-specific-condition-keys)) | High | 1 | 1 | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Low | 2 | - | - |

#### Intercept data in transit to an external bucket

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T12 | | **Name** | Intercept data in transit to an external bucket | | **Description** | S3 allows communication over HTTP. An attacker can intercept the traffic you send to an external bucket, in order to read or modify the data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (4.6)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:P/AC:H/PR:N/UI:R/S:U/C:H/I:L/A:N) | | **IAM Access** | {  "UNIQUE": "s3:any"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Enforce encryption-in-transit**    Block all unencrypted requests and unauthorized TLS version(s) from IAM entities you control (e.g. by denying all unencrypted requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" !=*authorized TLS version(s)*, using an SCP on your AWS Organization root node)    Block all unencrypted requests and unauthorized TLS version(s) from VPC endpoints you control (e.g. by denying all requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" != *authorized TLS version(s)*, on the VPC endpoint policy)    Monitor and investigate that all requests made with HTTP (e.g., via CloudTrail S3 data events with the lack of additionalEventData.CipherSuite)    Maintain a list of authorized version(s) of TLS/SSL per bucket (or per account/OU/Org) | Very High | 1 | 2 | 1 |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Medium | 1 | - | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Block S3 endpoints in your corporate perimeter security**    Block S3 endpoints ([DNS](https://docs.aws.amazon.com/general/latest/gr/s3.html) and [IP ranges](https://aws.amazon.com/premiumsupport/knowledge-center/s3-find-ip-address-ranges/)) in your corporate perimeter security to the Internet (e.g. firewalls, or cloud interception proxy like [Kivera](https://kivera.io)) including via Internet Gateway, to force usage of VPC endpoints. It will block data-plane transfer. Note: AWS console stays functional as it proxies non-data-plane requests (via "console.aws.amazon.com"). | Low | 1 | - | - |

#### Unauthorized object restored into an unauthorized bucket

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T26 | | **Name** | Unauthorized object restored into an unauthorized bucket | | **Description** | Objects can be stored in S3 Glacier. An attacker can restore an object to an unauthorized S3 bucket to collect or exfiltrate data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (4.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:R/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:RestoreObject"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

#### Upload in an authorized external bucket but an incorrect AWS account

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T31 | | **Name** | Upload in an authorized external bucket but an incorrect AWS account | | **Description** | Bucket names are globally unique. An attacker can take over a legitimate external bucket and deceive you into sending it to their bucket. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (4.0)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:H/UI:R/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutObject"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    Monitor that only authorized external buckets are used (e.g. via CloudTrail S3 data events in resources[].accountId and resources[].ARN). Both account ID and bucket name must be verified.    Request access via an S3 access point on a bucket you don't own, if compatible with your interaction with the bucket (e.g. not through not-compatible AWS service) | Very High | 2 | - | 1 |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Low | 1 | - | - |

#### Exfiltrate data via an ungoverned S3 endpoint

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T45 | | **Name** | Exfiltrate data via an ungoverned S3 endpoint | | **Description** | S3 VPC endpoints can be either Interface or Gateway. An attacker can create a second endpoint to create an ungoverned exfiltration vector. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0005](https://attack.mitre.org/tactics/TA0005) | | **CVSS** | [Low (1.9)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:U/C:N/I:L/A:N) | | **IAM Access** | {  "UNIQUE": "ec2:CreateVpcEndpoint"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Limit and monitor access via S3 VPC endpoints**    Ensure all S3 VPC endpoints (Interface and Gateway) are covered by the VPC endpoints controls | Very High | 1 | - | - |

#### Use of less secure or old S3 features

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T35 | | **Name** | Use of less secure or old S3 features | | **Description** | S3 was launched in 2006, and its features have evolved. An attacker can use older features that have been proven less secure by AWS (e.g. certain API configuration, [SigV2](https://aws.amazon.com/blogs/aws/amazon-s3-update-sigv2-deprecation-period-extended-modified/), path-style model), but are still maintained for retro-compatibility. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Low (1.9)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:U/C:N/I:L/A:N) | | **IAM Access** | {  "UNIQUE": "s3:deprecated"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Enforce good coding practice**    When connecting to S3 endpoints, use virtual-hosted model ("my-bucket-name.s3.amazonaws.com" or "my-bucket-name.my-s3-regional-endpoint.amazonaws.com") instead of path-style model ("s3.amazonaws.com/my-bucket-name" or "my-s3-regional-endpoint.amazonaws.com/my-bucket-name") (see [ref](https://aws.amazon.com/blogs/aws/amazon-s3-path-deprecation-plan-the-rest-of-the-story/)). All the latest SDKs make use of domain style by default.    Monitor that all S3 connections are made with the virtual-hosted model (e.g. via CloudTrail S3 requestParameters.Host) | Very High | 1 | - | 1 |
| **Block all requests not using SigV4**    Block all requests not using SigV4 (e.g. using an SCP and S3 policy on all buckets with deny on "StringNotEquals":{"s3:signatureversion": "AWS4-HMAC-SHA256"})    Monitor and investigate that all requests not using SigV4 (e.g. via CloudTrail S3 with the additionalEventData.SignatureVersion different from "SigV4")    Use SDK with SigV4 enabled ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingAWSSDK.html#UsingAWSSDK-move-to-Sig4)) | Very High | 1 | 1 | 1 |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | High | 2 | - | - |
| **Block deprecated actions**    Block deprecated S3 actions using IAM ThreatModel and the S3 actions list. | Medium | 1 | - | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bucket *(subclass of Object operations, FC5)* *To upload your data into your AWS account, you must create an S3 bucket in one of the AWS Regions.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Creates a new bucket | s3:CreateBucket |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Use a bucket to upload malware or modify an object to include malware | [High (7.3)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:L/UI:R/S:C/C:H/I:H/A:N) | | Exfiltrate data by using compromised IAM credentials from the Internet Internet | [Medium (6.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N) | | Files encrypted for ransomware | [Medium (6.3)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:L/UI:N/S:U/C:N/I:H/A:L) | | Object made public or accessible in a private bucket you own by changing object ACL | [Medium (5.9)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:R/S:U/C:H/I:H/A:N) | | Bucket takeover to gather data | [Medium (5.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:R/S:U/C:H/I:L/A:N) | | Intercept data in transit to an internal bucket | [Medium (4.6)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:P/AC:H/PR:N/UI:R/S:U/C:H/I:L/A:N) | | Use AWS services to access data on S3 | [Medium (4.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:H/UI:N/S:U/C:H/I:N/A:N) | | Move prod data in non-prod environment | [Medium (4.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:H/UI:N/S:U/C:H/I:N/A:N) | | Hotlinking content from S3 bucket | [Low (3.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:N/UI:R/S:U/C:N/I:L/A:N) | | Increase bill by restoring a large amount of data | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | Increase bill by creating incomplete uploads | [Low (2.3)](https://www.first.org/cvss/calculator/3.1#CVSS:3.0/AV:L/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | Abuse MD5 etag | [Low (1.8)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:H/PR:H/UI:R/S:U/C:N/I:L/A:N) | |

#### Use a bucket to upload malware or modify an object to include malware

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T14 | | **Name** | Use a bucket to upload malware or modify an object to include malware | | **Description** | S3 buckets are commonly used to distribute software. An attacker can upload malware in a bucket to better position it for later use or directly change an object to include malware ([example](https://www.securityweek.com/exposed-twilio-sdk-abused-malvertising-attack)). | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0001](https://attack.mitre.org/tactics/TA0001) | | **CVSS** | [High (7.3)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:L/UI:R/S:C/C:H/I:H/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutObject"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block direct public access**    Front buckets that are required to be public, using authenticated CDN (e.g. CloudFront) or API Gateway, protected with WAF (e.g. for [hotlinking](https://aws.amazon.com/blogs/security/how-to-prevent-hotlinking-by-using-aws-waf-amazon-cloudfront-and-referer-checking/))    Enable account-level S3 Block Public Access on all AWS accounts, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true.    Enable S3 Block Public Access on all S3 buckets, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true.    Enable S3 Block Public Access on all S3 access points (including multi-region), with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Very High | 4 | - | - |
| **Limit access from only authorized VPCs**    For each S3 bucket, maintain a list of VPC(s) authorized to access it.    Limit the access to only those VPC(s) (e.g. using S3 bucket statement, deny if the condition "aws:SourceVpce", or if the bucket policy size is beyond the limit, use this condition on access point) | High | 1 | 1 | - |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    Scan all data before uploading to an external bucket to ensure the classification of the data is aligned with the bucket classification (e.g. using Macie). | High | 2 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | High | 2 | - | - |
| **Scan input/output objects for malware**    If the bucket is used as an input or the output of a process, scan the objects for malware (e.g. using [VirusScan](https://s3-virusscan.widdix.net/), [Cloud Storage Security](https://cloudstoragesec.com/amazon-s3), or [Trend Micro Cloud One](https://aws.amazon.com/blogs/apn/amazon-s3-malware-scanning-using-trend-micro-cloud-one-and-aws-security-hub/)) | Medium | - | - | 1 |

#### Exfiltrate data by using compromised IAM credentials from the Internet Internet

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T39 | | **Name** | Exfiltrate data by using compromised IAM credentials from the Internet Internet | | **Description** | IAM credentials can be compromised (directly or using [pre-signed URL](https://docs.aws.amazon.com/AmazonS3/latest/dev/ShareObjectPreSignedURL.html)). An attacker can use a compromised but authorized IAM credential to download your object from an internal bucket via the public endpoint (using or not their own VPC endpoint). | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (6.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:GetObject"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Limit access from only authorized VPCs**    For each S3 bucket, maintain a list of VPC(s) authorized to access it.    Limit the access to only those VPC(s) (e.g. using S3 bucket statement, deny if the condition "aws:SourceVpce", or if the bucket policy size is beyond the limit, use this condition on access point) | Very High | 1 | 1 | - |
| **Block all requests not using HTTP authorization header, if not explicitly authorized**    Block all requests not using HTTP authorization header, i.e. presign via query strings or POST ([ref](https://docs.aws.amazon.com/AmazonS3/latest/API/sig-v4-authenticating-requests.html)) (e.g. using an SCP and S3 policy on all buckets with deny on "StringNotEquals":{"s3:authType": "REST-HEADER"}). Note that it blocks uploads via the console, as well.    Monitor and investigate that all requests not using SigV4 (e.g. via CloudTrail S3 with the additionalEventData.SignatureVersion different from "SigV4") | Medium | - | 1 | 1 |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

#### Files encrypted for ransomware

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T16 | | **Name** | Files encrypted for ransomware | | **Description** | S3 provides several types of encryption where the key is not operated by AWS (e.g. SSE-KMS with Bring Your Own Key). An attacker can encrypt all the data stored in S3 to ransom the data owner to get the decryption key ([blog](https://rhinosecuritylabs.com/aws/s3-ransomware-part-1-attack-vector)). Alternatively, an attacker can change the default encryption key, for a similar effect on any new data uploaded. | | **Goal** | Direct Financial Gain | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Medium (6.3)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:L/UI:N/S:U/C:N/I:H/A:L) | | **IAM Access** | {  "AND": ["s3:GetObject", "s3:PutObject"]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Enforce encryption-at-rest**    Maintain a list of authorized KMS key(s) for each bucket and their default encryption key. You might simplify by using only 1 key per bucket, ideally dedicated. Note that an S3 server access log bucket does not support KMS encryption ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/bucket-encryption.html#bucket-encryption-how-to-set-up)).    Ensure all objects on S3 buckets are encrypted with an authorized KMS key    Block PutObject requests with unauthorized KMS key on each bucket (e.g. using an S3 bucket policy deny statement on PutObject if the condition if exists "s3:x-amz-server-side-encryption-aws-kms-key-id" is not an authorized KMS key)    Monitor that only authorized KMS key(s) are used on each bucket (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*)    Maintain a list of buckets (or paths) required to be encrypted with server-side encryption with customer-provided keys (SSE-C)    For buckets (or paths) requiring SSE-C, block PutObject requests with unauthorized encryption (e.g. using an S3 bucket policy deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-customer-algorithm"="AES265" is not present)    For buckets (or paths) requiring SSE-C, monitor that only authorized encryption is used on each bucket or path (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-side-encryption-customer-algorithm*) | Very High | 3 | 2 | 2 |
| **Protect primary data against loss**    Enable versioning on buckets holding primary data    Backup primary data in a secure location under a different security authority (e.g. in an [AWS data bunker account](https://wellarchitectedlabs.com/security/100_labs/100_create_a_data_bunker/1_instructions/) via replication, or using AWS Backup for Amazon S3) | Very High | 2 | - | - |
| **Use S3 Object Lock to protect data integrity**    Implement the authorized default S3 Object Lock on each bucket (note: Amazon S3 evaluates and applies bucket policies before applying bucket default S3 Object Lock settings)    Block PutObject and PutObjectRetention requests with unauthorized S3 Object Lock on each bucket (e.g. using an S3 bucket policy deny statement on PutObject and PutObjectRetention if the condition if exists "s3:object-lock-mode" and "s3:object-lock-remaining-retention-days" is not the defined S3 Object Lock configuration) | Very High | - | 2 | - |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Monitor S3 with Amazon GuardDuty and Macie**    Enable and monitor [S3 protection in Amazon GuardDuty](https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_finding-types-s3.html) in all AWS accounts in all Regions, and protect it using GuardDuty ThreatModel. Ensure findings are investigated (e.g. using Amazon Detective). | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |
| **Have a process to apply legal hold**    Create a process to apply a legal hold to any S3 bucket whenever required. The condition "s3:object-lock-legal-hold" can be used to restrict who can remove such a lock. | Low | 1 | - | - |

#### Object made public or accessible in a private bucket you own by changing object ACL

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T36 | | **Name** | Object made public or accessible in a private bucket you own by changing object ACL | | **Description** | Bucket authority only prevails on object ACL when the object access is explicitly denied by the bucket authority ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/access-control-auth-workflow-object-operation.html)). An attacker (or someone by negligence) can change the object ACL to make it public or accessible for themselves (to exfiltrate or modify). | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.9)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:R/S:U/C:H/I:H/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutObjectAcl"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block changes to make an object public via object ACL**    Deny requests to change object ACL to public (e.g. using an SCP, S3 bucket policy, and VPC endpoint policy blocking PutObjectAcl for "s3:x-amz-grant-read", "s3:x-amz-grant-read-acp", "s3:x-amz-grant-write-acp", "s3:x-amz-grant-full-control" on the following predefined groups "http://acs.amazonaws.com/groups/global/AllUsers" and "http://acs.amazonaws.com/groups/global/AuthenticatedUsers")    Monitor ObjectACL changed (or tentatively changed) to public using CloudTrail S3 data events    Monitor and investigate anonymous requests to objects (e.g. using CloudTrail S3 data events with userIdentity.accountId=ANONYMOUS\_PRINCIPAL) | Very High | - | 1 | 2 |
| **Block direct public access**    Enable account-level S3 Block Public Access on all AWS accounts, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true.    Enable S3 Block Public Access on all S3 buckets, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true.    Enable S3 Block Public Access on all S3 access points (including multi-region), with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Very High | 3 | - | - |
| **Enforce encryption-at-rest**    Maintain a list of authorized KMS key(s) for each bucket and their default encryption key. You might simplify by using only 1 key per bucket, ideally dedicated. Note that an S3 server access log bucket does not support KMS encryption ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/bucket-encryption.html#bucket-encryption-how-to-set-up)).    Ensure all objects on S3 buckets are encrypted with an authorized KMS key    Use KMS ThreatModel to protect the KMS keys used for S3 (e.g. using encryptionContext on the policy of each KMS key)    Implement an authorized default encryption key on each bucket; and enable S3 Bucket Key, if CloudTrail events are not required for KMS encrypt/decrypt (note: Amazon S3 evaluates and applies bucket policies before applying bucket default encryption settings)    Block PutObject requests with unauthorized KMS key on each bucket (e.g. using an S3 bucket policy deny statement on PutObject if the condition if exists "s3:x-amz-server-side-encryption-aws-kms-key-id" is not an authorized KMS key)    Monitor that only authorized KMS key(s) are used on each bucket (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*)    Maintain a list of buckets (or paths) required to be encrypted with server-side encryption with customer-provided keys (SSE-C)    For buckets (or paths) requiring SSE-C, block PutObject requests with unauthorized encryption (e.g. using an S3 bucket policy deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-customer-algorithm"="AES265" is not present)    For buckets (or paths) requiring SSE-C, monitor that only authorized encryption is used on each bucket or path (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-side-encryption-customer-algorithm*) | Very High | 5 | 2 | 2 |
| **Limit access from only authorized VPCs**    For each S3 bucket, maintain a list of VPC(s) authorized to access it.    Limit the access to only those VPC(s) (e.g. using S3 bucket statement, deny if the condition "aws:SourceVpce", or if the bucket policy size is beyond the limit, use this condition on access point) | Very High | 1 | 1 | - |
| **Disabling ACLs for all buckets**    Ensure bucket ACL and object ACL are disabled on each bucket    Prevent the creation of buckets with ACL enabled (e.g. by using a SCP and/or an IAM policy on "s3:CreateBucket" with a deny statement on StringNotEquals "s3:x-amz-object-ownership":"BucketOwnerEnforced"). Note that it does not block someone from enabling an ACL afterward via PutPutBucketOwnershipControls. | Very High | 1 | 1 | - |
| **Monitor S3 with Amazon GuardDuty and Macie**    Enable [S3 policy findings in Amazon Macie](https://docs.aws.amazon.com/macie/latest/user/findings-types.html#findings-policy-types) in all AWS accounts in all Regions, and protect it using Macie ThreatModel | High | 1 | - | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Low | 2 | - | - |

#### Bucket takeover to gather data

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T1 | | **Name** | Bucket takeover to gather data | | **Description** | Bucket names are globally unique and can be recreated after 1 hour from deletion in another AWS account. An attacker can recreate the same bucket name of a deleted bucket you used to own to collect any new data uploaded by a non-updated party, do a DNS takeover (using a non-deleted CNAME / CloudFront origin to the bucket), or use remaining permissions to exfiltrate data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:R/S:U/C:H/I:L/A:N) | | **IAM Access** | {  "OPTIONAL": "s3:DeleteBucket"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Prevent deletion of buckets**    Block the action "s3:DeleteBucket" (e.g. via SCP, exemption can be managed by authorizing a SuperAdmin to delete buckets with a certain tag, and with bucket owners able to tag bucket)    Scan your CNAME records (e.g. in Route53) and CloudFront origin for deleted buckets | Very High | - | 1 | 1 |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Enforce good coding practice**    Parameterize the S3 bucket name or S3 access point in your code (no hardcoding)    Ensure all S3 buckets interacted with are in the correct AWS account (e.g. using the condition in all compatible S3 requests: x-amz-expected-bucket-owner and x-amz-source-expected-bucket-owner) | Medium | 2 | - | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |
| **Identify and ensure the protection of all external buckets hosting your objects**    Monitor that only authorized external buckets are used (e.g. via CloudTrail S3 data events in resources[].accountId and resources[].ARN). Both account ID and bucket name must be verified. | Low | - | - | 1 |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Very Low | 1 | - | - |

#### Intercept data in transit to an internal bucket

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T34 | | **Name** | Intercept data in transit to an internal bucket | | **Description** | S3 allows communication over HTTP. An attacker can intercept the traffic you send on an internal bucket, in order to read or modify the data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (4.6)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:P/AC:H/PR:N/UI:R/S:U/C:H/I:L/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Enforce encryption-in-transit**    Block all unencrypted requests and unauthorized TLS version(s) from IAM entities you control (e.g. by denying all unencrypted requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" !=*authorized TLS version(s)*, using an SCP on your AWS Organization root node)    Block all unencrypted requests and unauthorized TLS version(s) from VPC endpoints you control (e.g. by denying all requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" != *authorized TLS version(s)*, on the VPC endpoint policy)    Monitor and investigate that all requests made with HTTP (e.g., via CloudTrail S3 data events with the lack of additionalEventData.CipherSuite)    Block all unencrypted requests to S3 bucket you control (e.g. by denying all requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" != *authorized TLS version(s)*, on the S3 bucket policy) | Very High | - | 3 | 1 |
| **Block S3 endpoints in your corporate perimeter security**    Block S3 endpoints ([DNS](https://docs.aws.amazon.com/general/latest/gr/s3.html) and [IP ranges](https://aws.amazon.com/premiumsupport/knowledge-center/s3-find-ip-address-ranges/)) in your corporate perimeter security to the Internet (e.g. firewalls, or cloud interception proxy like [Kivera](https://kivera.io)) including via Internet Gateway, to force usage of VPC endpoints. It will block data-plane transfer. Note: AWS console stays functional as it proxies non-data-plane requests (via "console.aws.amazon.com"). | Very High | 1 | - | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |

#### Use AWS services to access data on S3

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T30 | | **Name** | Use AWS services to access data on S3 | | **Description** | Number of AWS services can access S3 to execute their functions. An attacker can use them to collect data, using their service role or service-linked roles. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0009](https://attack.mitre.org/tactics/TA0009) | | **CVSS** | [Medium (4.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:H/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "iam:PassRole"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Enforce encryption-at-rest**    Maintain a list of authorized KMS key(s) for each bucket and their default encryption key. You might simplify by using only 1 key per bucket, ideally dedicated. Note that an S3 server access log bucket does not support KMS encryption ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/bucket-encryption.html#bucket-encryption-how-to-set-up)).    Ensure all objects on S3 buckets are encrypted with an authorized KMS key    Block PutObject requests with unauthorized KMS key on each bucket (e.g. using an S3 bucket policy deny statement on PutObject if the condition if exists "s3:x-amz-server-side-encryption-aws-kms-key-id" is not an authorized KMS key)    Monitor that only authorized KMS key(s) are used on each bucket (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*)    Maintain a list of buckets (or paths) required to be encrypted with server-side encryption with customer-provided keys (SSE-C)    For buckets (or paths) requiring SSE-C, block PutObject requests with unauthorized encryption (e.g. using an S3 bucket policy deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-customer-algorithm"="AES265" is not present)    For buckets (or paths) requiring SSE-C, monitor that only authorized encryption is used on each bucket or path (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-side-encryption-customer-algorithm*) | Very High | 3 | 2 | 2 |
| **Limit access from only authorized VPCs**    For each S3 bucket, maintain a list of VPC(s) authorized to access it.    Limit the access to only those VPC(s) (e.g. using S3 bucket statement, deny if the condition "aws:SourceVpce", or if the bucket policy size is beyond the limit, use this condition on access point) | Very High | 1 | 1 | - |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Model the threats on all AWS services accessing S3**    Analyze and protect all AWS services accessing S3 (e.g. via ThreatModel). Enforce usage in VPC only, whenever possible. | High | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Low | 1 | - | - |

#### Move prod data in non-prod environment

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T11 | | **Name** | Move prod data in non-prod environment | | **Description** | Multiple types of environments are usually operated in AWS. An attacker can move the data from a secure location (e.g. production) to a less secure location (e.g. dev). | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0009](https://attack.mitre.org/tactics/TA0009) | | **CVSS** | [Medium (4.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:H/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:GetObject"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Identify and ensure the protection of all internal buckets hosting your objects**    Track all buckets you control, define their authorized data classification, identify whether the hosted data is primary (i.e. source of truth, for example logs, backups, forensic data, raw data, etc.) or an input/output of a process (e.g. file-processing, software package, etc.), their WORM requirements (e.g. SEC 17a-4, CTCC, etc.), if they are production/non-production (preferably done at account-level), their storage class. You may use tags, Infra-as-code, AWS Glue Data Catalog, or external management tools like [FINRA herd](https://finraos.github.io/herd/))    Use a data discovery tool (e.g. Amazon Macie) to ensure no sensitive data is stored in an unauthorized bucket    Use a data discovery tool (e.g. Amazon Macie) to ensure the bucket names, object names, tags, and metadata do not contain sensitive data | Very High | 1 | - | 2 |
| **Enforce encryption-at-rest**    Maintain a list of authorized KMS key(s) for each bucket and their default encryption key. You might simplify by using only 1 key per bucket, ideally dedicated. Note that an S3 server access log bucket does not support KMS encryption ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/bucket-encryption.html#bucket-encryption-how-to-set-up)).    Ensure all objects on S3 buckets are encrypted with an authorized KMS key    Block PutObject requests with unauthorized KMS key on each bucket (e.g. using an S3 bucket policy deny statement on PutObject if the condition if exists "s3:x-amz-server-side-encryption-aws-kms-key-id" is not an authorized KMS key)    Monitor that only authorized KMS key(s) are used on each bucket (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*)    Maintain a list of buckets (or paths) required to be encrypted with server-side encryption with customer-provided keys (SSE-C)    For buckets (or paths) requiring SSE-C, block PutObject requests with unauthorized encryption (e.g. using an S3 bucket policy deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-customer-algorithm"="AES265" is not present)    For buckets (or paths) requiring SSE-C, monitor that only authorized encryption is used on each bucket or path (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-side-encryption-customer-algorithm*) | Very High | 3 | 2 | 2 |
| **Restrict access point access to VPC when in use**    Maintain a list of authorized access between VPCs, S3 access points, and S3.    Limit access via the S3 access point by using a VPC endpoint and/or bucket policy with the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" in an allow statement to reduce the length of the allowlist bucket name in the VPC endpoint/bucket policy.    Block any [object-related operations](https://docs.aws.amazon.com/AmazonS3/latest/dev/using-access-points.html#access-points-service-api-support) access to S3 bucket not through access point (i.e. using a deny IAM policy statement with the condition "ArnNotLike" {"s3:DataAccessPointArn": "arn:aws:s3:*Region*:*AccountId*:accesspoint/\*"}) | Very High | 1 | 2 | - |
| **Limit and monitor access via S3 VPC endpoints**    For each VPC, maintain a list of AWS Organizations, OU, and/or AWS account(s) where IAM entities are authorized to access S3    For each VPC with an IAM entity allowed to use S3, secure them with the VPC ThreatModel (e.g. [modification of VPC endpoints, VPC endpoint policy, routing table, Security Groups](https://docs.aws.amazon.com/vpc/latest/userguide/vpce-gateway.html))    Block any IAM entity not belonging to an authorized AWS Organizations, OU, and/or AWS account(s) to call S3 from your VPCs by adding a deny statement on the S3 VPC endpoint policy of each VPC, with the condition using "aws:PrincipalOrgPaths" ([ref](https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_condition-keys.html#condition-keys-principalorgpaths)) including the full Org ID, as those are globally unique.    Enable [VPC DNS query logging](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resolver-query-logs.html) in all VPC    Maintain a list of authorized S3 and S3 access point (and their respective AWS accounts) to be accessed for each VPC    Limit the access to only authorized S3 bucket(s) or their AWS account(s) from each VPC (e.g. using the condition key "s3:ResourceAccount" on the VPC endpoint policy, alternatively use a specific resource-level statement for each bucket, or if the VPC endpoint policy size is beyond the limit and more granular control on VPC is required, use access points)    Monitor VPC DNS query logs that only authorized S3 bucket and S3 access points are being queried in each VPC (e.g. using VPC DNS query logging), and protect it using Route53 ThreatModel | Very High | 4 | 2 | 1 |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Limit access from only authorized VPCs**    For each S3 bucket, maintain a list of VPC(s) authorized to access it.    Limit the access to only those VPC(s) (e.g. using S3 bucket statement, deny if the condition "aws:SourceVpce", or if the bucket policy size is beyond the limit, use this condition on access point) | High | 1 | 1 | - |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    Monitor that only authorized external buckets are used (e.g. via CloudTrail S3 data events in resources[].accountId and resources[].ARN). Both account ID and bucket name must be verified. | High | 1 | - | 1 |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Medium | 1 | - | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

#### Hotlinking content from S3 bucket

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T22 | | **Name** | Hotlinking content from S3 bucket | | **Description** | S3 charges for hosting and data transfer out. An attacker can hotlink your content hosted on S3 on another page to avoid paying the S3 bills ([ref](https://aws.amazon.com/blogs/security/how-to-prevent-hotlinking-by-using-aws-waf-amazon-cloudfront-and-referer-checking/)). | | **Goal** | Financial Drain | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Low (3.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:N/UI:R/S:U/C:N/I:L/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block direct public access**    Front buckets that are required to be public, using authenticated CDN (e.g. CloudFront) or API Gateway, protected with WAF (e.g. for [hotlinking](https://aws.amazon.com/blogs/security/how-to-prevent-hotlinking-by-using-aws-waf-amazon-cloudfront-and-referer-checking/)) | Very High | 1 | - | - |
| **Monitor S3 with Amazon GuardDuty and Macie**    Enable [S3 policy findings in Amazon Macie](https://docs.aws.amazon.com/macie/latest/user/findings-types.html#findings-policy-types) in all AWS accounts in all Regions, and protect it using Macie ThreatModel | High | 1 | - | - |

#### Increase bill by restoring a large amount of data

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T47 | | **Name** | Increase bill by restoring a large amount of data | | **Description** | Restore cost can be amplified by the size and the type (i.e. expedited). An attacker can restore lots of data to generate costs. | | **Goal** | Financial Drain | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | **IAM Access** | {  "UNIQUE": "s3:RestoreObject"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Medium | 1 | - | - |

#### Increase bill by creating incomplete uploads

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T40 | | **Name** | Increase bill by creating incomplete uploads | | **Description** | By default, when a multiple upload is initiated but not completed, S3 will keep it ([ref](https://www.reddit.com/r/aws/comments/immer3/protip_watch_out_for_stranded_multipart_uploads_i/)). An attacker can upload a large amount of data without completing it while being hard to detect. | | **Goal** | Financial Drain | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Low (2.3)](https://www.first.org/cvss/calculator/3.1#CVSS:3.0/AV:L/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutObject"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Remove incomplete multipart uploads**    Reduce costs related to incomplete multipart upload by creating a lifecycle policy to remove them after an agreed length of time (e.g. 7 days) ([blog](https://aws.amazon.com/blogs/aws/s3-lifecycle-management-update-support-for-multipart-uploads-and-delete-markers/)) | Very High | - | 1 | - |

#### Abuse MD5 etag

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T27 | | **Name** | Abuse MD5 etag | | **Description** | Etags include the MD5 of the file but not consistently and can be used by developers to verify the integrity of a file. An attacker can affect an upload function to change the etag of a file to disrupt a workflow downstream. | | **Goal** | Data manipulation | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Low (1.8)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:H/PR:H/UI:R/S:U/C:N/I:L/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Enforce good coding practice**    Use "x-amz-checksum" from the [object metadata](https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingMetadata.html#SysMetadata) to validate the integrity of the object instead of etag. If etag is used, make sure properly account for its different definitions ([ref](https://teppen.io/2018/06/23/aws_s3_etags/)) | Very High | 1 | - | - |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Object tagging *(subclass of Object operations, used by Bucket, FC2)* *You can tag objects (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/object-tagging.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Adds a set of tags to an existing object. | s3:PutObjectTagging |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Gain access by modifying or deleting important object tags | [Medium (4.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:L/UI:N/S:C/C:L/I:L/A:N) | |

#### Gain access by modifying or deleting important object tags

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T33 | | **Name** | Gain access by modifying or deleting important object tags | | **Description** | Tags can be used for various reasons, including security classification or access management (via ABAC). An attacker can change the tagging of an object to another value enabling them to execute another attack. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0004](https://attack.mitre.org/tactics/TA0004) | | **CVSS** | [Medium (4.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:L/UI:N/S:C/C:L/I:L/A:N) | | **IAM Access** | {  "OR": ["s3:PutObjectTagging", "s3:DeleteObjectTagging"]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Limit access from only authorized VPCs**    For each S3 bucket, maintain a list of VPC(s) authorized to access it.    Limit the access to only those VPC(s) (e.g. using S3 bucket statement, deny if the condition "aws:SourceVpce", or if the bucket policy size is beyond the limit, use this condition on access point) | Very High | 1 | 1 | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | High | 2 | - | - |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Torrent *(subclass of Object operations, used by Bucket, FC21)* ***[NOT RECOMMENDED]*** *You can use the BitTorrent protocol to retrieve objects (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/S3Torrent.html)*). Only available in the AWS Regions launched before May 30, 2016. The seed rate is 100KB. After April 29, 2022, BitTorrent clients will no longer connect to Amazon S3.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Returns torrent files from an object. | s3:GetObjectTorrent |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Batch *(subclass of Object operations, used by Bucket, FC27)* *S3 Batch Operations performs large-scale Batch Operations on Amazon S3 objects.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Creates a new Amazon S3 Batch Operations job. | s3:CreateJob |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Exfiltrate, modify or delete objects using Batch | [Medium (6.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:H/UI:N/S:C/C:H/I:L/A:N) | |

#### Exfiltrate, modify or delete objects using Batch

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T44 | | **Name** | Exfiltrate, modify or delete objects using Batch | | **Description** | S3 Batch Operations require an IAM role (with proper trust policy), then can run operations including replicating existing objects, copying, or replacing/deleting object tags. An attacker can use Batch copy or modify objects to exfiltrate or change the access management of an object (if relying on a tag). | | **Goal** | Data manipulation | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Medium (6.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:H/UI:N/S:C/C:H/I:L/A:N) | | **IAM Access** | {  "AND": ["s3:CreateJob", "iam:PassRole"]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Control IAM roles used for Batch**    Maintain a list of IAM roles used for Batch job, ideally dedicated (e.g. using change management process on infrastructure-as-code)    Ensure only an authorized IAM role is attached on each Batch job    Limit the access to only required resources/permissions (e.g. source/destination bucket, Lambda functions) of each authorized IAM role configured for Batch jobs    Limit access to authorized IAM roles used for Batch job, using the IAM ThreatModel (e.g. trust policy, and "iam:PassRole") | Very High | 4 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Object versioning *(subclass of Object operations, used by Bucket, FC3)* *You can version your objects (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/ObjectVersioning.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Retrieves an object version from Amazon S3. | s3:GetObjectVersion |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tag on versioned objects *(subclass of Object tagging/Object versioning, used by Bucket, FC4)* *You can tag a specific version of an object (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/object-tagging.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Adds a set of tags to an existing object version | s3:PutObjectVersionTagging |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ACL on versioned objects *(subclass of Object versioning/ACL on versioned objects, FC9)* *Amazon S3 Access Control Lists (ACLs) enable you to manage access to object versions (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#permissions)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Sets the Access Control List (ACL) permissions for an object version. You must have WRITE\_ACP permission to set the ACL of an object version. | s3:PutObjectVersionAcl |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bucket versioning *(subclass of Object versioning/Bucket, FC6)* *Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from unintended user actions and application failures (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/Versioning.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Sets the versioning state of an existing bucket. | s3:PutBucketVersioning |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Affect data protection by removing versioning | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | |

#### Affect data protection by removing versioning

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T48 | | **Name** | Affect data protection by removing versioning | | **Description** | Versioning can be used as the first level of integrity protection. An attacker can suspend versioning to affect the data protection of a bucket. | | **Goal** | Data manipulation | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutBucketVersioning"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Medium | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Replication *(subclass of Bucket versioning, FC15)* *Replication enables automatic and asynchronous copying of objects of a bucket into another bucket. It can be cross-region or in the same region. Buckets configured for replication can be in the same AWS account or different accounts. It is usually to backup S3 data, data centralization, or multi-region applications.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Creates a new replication configuration (or replaces an existing one, if present). | s3:PutReplicationConfiguration |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Unauthorized access to data via bucket replication | [Medium (4.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:H/I:N/A:N) | | Affect data protection by removing replication | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | |

#### Unauthorized access to data via bucket replication

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T2 | | **Name** | Unauthorized access to data via bucket replication | | **Description** | Replication allows you to replicate objects, their metadata and change ownership. The configuration focuses on new objects only (old objects replication requires using S3 Batch Replication). An attacker can configure replication on a bucket to replicate objects (or its metadata or tagging) in a bucket they control to exfiltrate data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (4.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "AND": ["s3:PutReplicationConfiguration", "iam:PassRole"]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Restrict bucket replication**    Maintain a list of authorized buckets to have replication enabled, their target bucket and replication type (i.e. ownership, RTC, etc.) ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/replication-add-config.html)).    Ensure only authorized buckets have replication enabled and with correct configuration are configured    Maintain a list of IAM roles used for replication, ideally dedicated (e.g. using change management process on infrastructure-as-code)    Ensure only authorized IAM roles are attached for each replication, ideally dedicated    Limit the S3 access to the source/destination bucket and replication rights of each authorized IAM role configured for replication    Limit access to authorized IAM roles used for replication, using the IAM ThreatModel (e.g. trust policy, and "iam:PassRole")    Monitor abnormal behavior on replication CloudWatch metrics (i.e. *BytesPendingReplication* and *OperationsPendingReplication*) | High | 6 | - | 1 |
| **Monitor S3 with Amazon GuardDuty and Macie**    Enable [S3 policy findings in Amazon Macie](https://docs.aws.amazon.com/macie/latest/user/findings-types.html#findings-policy-types) in all AWS accounts in all Regions, and protect it using Macie ThreatModel | High | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

#### Affect data protection by removing replication

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T49 | | **Name** | Affect data protection by removing replication | | **Description** | Replication can be used as a level of integrity protection and backup. An attacker can remove replication to affect the data protection. | | **Goal** | Data manipulation | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutReplicationConfiguration"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Medium | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bucket tag *(subclass of Bucket, FC7)* *You can tag buckets (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/CostAllocTagging.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Adds a set of tags to an existing bucket. | s3:PutBucketTagging |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Exfiltrate data by using tags | [Low (3.3)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N) | |

#### Exfiltrate data by using tags

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T18 | | **Name** | Exfiltrate data by using tags | | **Description** | Objects and buckets can have tags. An attacker can use those features to exfiltrate data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Low (3.3)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N) | | **IAM Access** | {  "AND": [{  "OR": ["GetObjectTagging", "s3:GetObjectVersionTagging"]  }, {  "OR": ["s3:PutObjectTagging", "s3:PutObjectVersionTagging"]  }]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block S3 endpoints in your corporate perimeter security**    Block S3 endpoints ([DNS](https://docs.aws.amazon.com/general/latest/gr/s3.html) and [IP ranges](https://aws.amazon.com/premiumsupport/knowledge-center/s3-find-ip-address-ranges/)) in your corporate perimeter security to the Internet (e.g. firewalls, or cloud interception proxy like [Kivera](https://kivera.io)) including via Internet Gateway, to force usage of VPC endpoints. It will block data-plane transfer. Note: AWS console stays functional as it proxies non-data-plane requests (via "console.aws.amazon.com"). | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bucket ACL *(subclass of Bucket, FC8)* ***[NOT RECOMMENDED]*** *Amazon S3 Access Control Lists (ACLs) enable you to manage access to buckets. Each bucket has an ACL attached to it as a sub-resource. It defines which AWS accounts or groups are granted access and the type of access. When a request is received against a resource, Amazon S3 checks the corresponding ACL to ensure the requester has the necessary access permissions (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#permissions)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Sets the permissions on an existing bucket using Access Control Lists (ACL). | s3:PutBucketAcl |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Grant unauthorized access to a private bucket by changing bucket ACL | [Medium (5.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:L/I:H/A:N) | | DoS by blocking traffic using bucket ACL | [Low (2.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:R/S:U/C:N/I:L/A:N) | |

#### Grant unauthorized access to a private bucket by changing bucket ACL

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T4 | | **Name** | Grant unauthorized access to a private bucket by changing bucket ACL | | **Description** | Bucket ACL can be used to give access to the bucket information, list the objects, and overwrite/delete objects. An attacker can change the bucket ACL to destroy or modify data, or exfiltrate data via the object name (1KB). | | **Goal** | Data manipulation | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Medium (5.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:L/I:H/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutBucketAcl"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block direct public access**    Enable account-level S3 Block Public Access on all AWS accounts, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true.    Enable S3 Block Public Access on all S3 buckets, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Very High | 2 | - | - |
| **Block bucket ACL**    Deny requests to add bucket ACL (e.g. using an SCP, bucket policy, and VPC endpoint policy blocking "s3:PutBucketAcl")    Monitor changes on bucket ACL to ensure it stays private (e.g. using CloudTrail event PutBucketAcl and its fields requestParameters.x-amz-acl should be either "private" or not existing) | Very High | - | 1 | 1 |
| **Disabling ACLs for all buckets**    Ensure bucket ACL and object ACL are disabled on each bucket    Prevent the creation of buckets with ACL enabled (e.g. by using a SCP and/or an IAM policy on "s3:CreateBucket" with a deny statement on StringNotEquals "s3:x-amz-object-ownership":"BucketOwnerEnforced"). Note that it does not block someone from enabling an ACL afterward via PutPutBucketOwnershipControls. | Very High | 1 | 1 | - |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Monitor S3 with Amazon GuardDuty and Macie**    Enable and monitor [S3 protection in Amazon GuardDuty](https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_finding-types-s3.html) in all AWS accounts in all Regions, and protect it using GuardDuty ThreatModel. Ensure findings are investigated (e.g. using Amazon Detective).    Enable [S3 policy findings in Amazon Macie](https://docs.aws.amazon.com/macie/latest/user/findings-types.html#findings-policy-types) in all AWS accounts in all Regions, and protect it using Macie ThreatModel | High | 2 | - | - |
| **Enable CloudTrail S3 data events**    Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Medium | 1 | - | - |

#### DoS by blocking traffic using bucket ACL

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T50 | | **Name** | DoS by blocking traffic using bucket ACL | | **Description** | Bucket ACL can allow access (e.g. for CloudFront access logs). An attacker can remove an existing permission to deny legitimate access to the bucket. | | **Goal** | Disruption of Service | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Low (2.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:R/S:U/C:N/I:L/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutBucketAcl"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Medium | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bucket policy *(subclass of Bucket, FC10)* *For your bucket, you can add a bucket policy to grant other AWS accounts or IAM users permissions for the bucket and the objects in it. Any object permissions apply only to the objects that the bucket owner creates.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Adds to or replaces a policy on a bucket. | s3:PutBucketPolicy |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Grant unauthorized access to a private bucket by changing bucket policy | [High (7.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:H/I:H/A:H) | | Reduce bucket security by deleting the bucket policy | [Medium (6.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:H/UI:N/S:U/C:H/I:H/A:H) | | Use CloudFront to access private bucket | [Medium (5.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | |

#### Grant unauthorized access to a private bucket by changing bucket policy

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T37 | | **Name** | Grant unauthorized access to a private bucket by changing bucket policy | | **Description** | Bucket policy can enable access to objects owned by the bucket. An attacker (or someone by negligence) can change the bucket policy and make the content accessible. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [High (7.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:H/I:H/A:H) | | **IAM Access** | {  "UNIQUE": "s3:PutBucketPolicy"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block direct public access**    Enable account-level S3 Block Public Access on all AWS accounts, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true.    Enable S3 Block Public Access on all S3 buckets, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true.    Enable S3 Block Public Access on all S3 access points (including multi-region), with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Very High | 3 | - | - |
| **Enforce encryption-at-rest**    Maintain a list of authorized KMS key(s) for each bucket and their default encryption key. You might simplify by using only 1 key per bucket, ideally dedicated. Note that an S3 server access log bucket does not support KMS encryption ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/bucket-encryption.html#bucket-encryption-how-to-set-up)).    Ensure all objects on S3 buckets are encrypted with an authorized KMS key    Use KMS ThreatModel to protect the KMS keys used for S3 (e.g. using encryptionContext on the policy of each KMS key)    Implement an authorized default encryption key on each bucket; and enable S3 Bucket Key, if CloudTrail events are not required for KMS encrypt/decrypt (note: Amazon S3 evaluates and applies bucket policies before applying bucket default encryption settings)    Block PutObject requests with unauthorized KMS key on each bucket (e.g. using an S3 bucket policy deny statement on PutObject if the condition if exists "s3:x-amz-server-side-encryption-aws-kms-key-id" is not an authorized KMS key)    Maintain a list of buckets (or paths) required to be encrypted with server-side encryption with customer-provided keys (SSE-C)    For buckets (or paths) requiring SSE-C, block PutObject requests with unauthorized encryption (e.g. using an S3 bucket policy deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-customer-algorithm"="AES265" is not present) | Very High | 5 | 2 | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account.    For each bucket, maintain a list of authorized IAM principals allowed to access via bucket policy    Ensure only authorized a list of authorized IAM principals allowed to access via bucket policy are configured (e.g. using IAM Access Analyzer for the reconciliation) | Very High | 4 | - | - |
| **Monitor S3 with Amazon GuardDuty and Macie**    Enable [S3 policy findings in Amazon Macie](https://docs.aws.amazon.com/macie/latest/user/findings-types.html#findings-policy-types) in all AWS accounts in all Regions, and protect it using Macie ThreatModel | High | 1 | - | - |

#### Reduce bucket security by deleting the bucket policy

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T38 | | **Name** | Reduce bucket security by deleting the bucket policy | | **Description** | Bucket policy can deny access to objects, as it supersedes the object authority. An attacker (or someone by negligence) can delete the bucket policy and make the content less secure. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0004](https://attack.mitre.org/tactics/TA0004) | | **CVSS** | [Medium (6.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:H/UI:N/S:U/C:H/I:H/A:H) | | **IAM Access** | {  "UNIQUE": "s3:DeleteBucketPolicy"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Limit access from only authorized VPCs**    For each S3 bucket, maintain a list of VPC(s) authorized to access it.    Limit the access to only those VPC(s) (e.g. using S3 bucket statement, deny if the condition "aws:SourceVpce", or if the bucket policy size is beyond the limit, use this condition on access point) | Very High | 1 | 1 | - |
| **Block direct public access**    Enable account-level S3 Block Public Access on all AWS accounts, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true.    Enable S3 Block Public Access on all S3 buckets, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true.    Enable S3 Block Public Access on all S3 access points (including multi-region), with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | High | 3 | - | - |
| **Monitor S3 with Amazon GuardDuty and Macie**    Enable [S3 policy findings in Amazon Macie](https://docs.aws.amazon.com/macie/latest/user/findings-types.html#findings-policy-types) in all AWS accounts in all Regions, and protect it using Macie ThreatModel | High | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | High | 2 | - | - |

#### Use CloudFront to access private bucket

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T20 | | **Name** | Use CloudFront to access private bucket | | **Description** | CloudFront distributions can use S3 as their origin. An attacker can connect a CloudFront distribution to a private S3 bucket to get access to it. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutBucketPolicy"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Enforce encryption-at-rest**    Maintain a list of authorized KMS key(s) for each bucket and their default encryption key. You might simplify by using only 1 key per bucket, ideally dedicated. Note that an S3 server access log bucket does not support KMS encryption ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/bucket-encryption.html#bucket-encryption-how-to-set-up)).    Ensure all objects on S3 buckets are encrypted with an authorized KMS key    Implement an authorized default encryption key on each bucket; and enable S3 Bucket Key, if CloudTrail events are not required for KMS encrypt/decrypt (note: Amazon S3 evaluates and applies bucket policies before applying bucket default encryption settings)    Block PutObject requests with unauthorized KMS key on each bucket (e.g. using an S3 bucket policy deny statement on PutObject if the condition if exists "s3:x-amz-server-side-encryption-aws-kms-key-id" is not an authorized KMS key)    Maintain a list of buckets (or paths) required to be encrypted with server-side encryption with customer-provided keys (SSE-C)    For buckets (or paths) requiring SSE-C, block PutObject requests with unauthorized encryption (e.g. using an S3 bucket policy deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-customer-algorithm"="AES265" is not present) | Very High | 4 | 2 | - |
| **Control CloudFront access**    Maintain a list of authorized CloudFront distribution (via Origin Access Control) and associated bucket    Ensure only authorized CloudFront distributions are associated with their authorized bucket, and vice versa. | High | 2 | - | - |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Low | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S3 access logging *(subclass of Bucket ACL/Bucket policy, FC19)* *Server access logging provides detailed records for the requests made to a bucket. CloudTrail S3 data events are preferred, due to the more reliable delivery timing, consistency, supporting KMS encryption and S3 Object Lock (*[*full comparison*](https://www.netskope.com/blog/aws-s3-logjam-server-access-logging-vs-object-level-logging)*), however website endpoint is not recorded on S3 data events, some SIEM modules might be more featured with S3 access logs, and access logging is free beside storage.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Sets the logging parameters for a bucket. | s3:PutBucketLogging |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Evade detection by disabling S3 access logs via bucket policy change/removal | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | Evade detection by disabling S3 access logs via bucket ACL change | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | Evade detection by modifying S3 access logs | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | |

#### Evade detection by disabling S3 access logs via bucket policy change/removal

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T58 | | **Name** | Evade detection by disabling S3 access logs via bucket policy change/removal | | **Description** | S3 access logs can be used by SIEM to detect abnormal behaviors. An attacker can disable S3 access logs via bucket policy changes on the logging destination bucket to evade detection. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0005](https://attack.mitre.org/tactics/TA0005) | | **CVSS** | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | **IAM Access** | {  "OR": ["s3:PutBucketPolicy", "s3:DeleteBucketPolicy"]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block bucket ACL**    Deny requests to add bucket ACL (e.g. using an SCP, bucket policy, and VPC endpoint policy blocking "s3:PutBucketAcl")    Monitor changes on bucket ACL to ensure it stays private (e.g. using CloudTrail event PutBucketAcl and its fields requestParameters.x-amz-acl should be either "private" or not existing) | Very High | - | 1 | 1 |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

#### Evade detection by disabling S3 access logs via bucket ACL change

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T51 | | **Name** | Evade detection by disabling S3 access logs via bucket ACL change | | **Description** | S3 access logs can be used by SIEM to detect abnormal behaviors. An attacker can disable S3 access logs via bucket ACL changes on the logging destination bucket to evade detection. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0005](https://attack.mitre.org/tactics/TA0005) | | **CVSS** | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutBucketAcl"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Medium | 1 | - | - |

#### Evade detection by modifying S3 access logs

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T59 | | **Name** | Evade detection by modifying S3 access logs | | **Description** | S3 access logs can be used by SIEM to detect abnormal behaviors. An attacker can modify or disable S3 access logs to evade detection. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0005](https://attack.mitre.org/tactics/TA0005) | | **CVSS** | [Low (2.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:L/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutBucketLogging"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Enforce S3 access logging**    Monitor PutBucketLogging to detect bucket logging changes including deactivation and bucket change (i.e. using CloudTrail event *PutBucketLogging* and *requestParameter.BucketLoggingStatus* field to examine the lack of "LoggingEnabled" key, or an unauthorized bucket in requestParameters.BucketLoggingStatus.LoggingEnabled.TargetBucket) | Medium | - | - | 1 |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Analytics *(subclass of Bucket, FC11)* *You can analyze storage access patterns to decide the storage class (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/analytics-storage-class.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Adds an analytics configuration (identified by the analytics ID) to the bucket. | s3:PutAnalyticsConfiguration |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Inventory *(subclass of Bucket, FC12)* *You can create a report on your storage, including object metadata or versions (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/storage-inventory.html#storage-inventory-how-to-set-up)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Adds an inventory configuration (identified by the inventory ID) to the bucket | s3:PutInventoryConfiguration |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Exfiltrate data via inventory | [Low (2.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:L/I:N/A:N) | |

#### Exfiltrate data via inventory

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T42 | | **Name** | Exfiltrate data via inventory | | **Description** | Inventory sends the object names (i.e. keys) to any configured S3 bucket. An attacker can use the name of objects (1KB) to exfiltrate data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Low (2.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:L/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutBucketInventory"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Enforce good coding practice**    Do not include sensitive data in bucket names, access point names, object names, object metadata and tags. | Medium | 1 | - | - |
| **Control where the inventory is stored**    Maintain a list of authorized S3 buckets to receive S3 inventory of each bucket    Ensure only authorized S3 buckets are configured to receive S3 inventory for each bucket | Medium | 2 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Lifecycle *(subclass of Bucket, FC13)* *You can lifecycle your data to reduce the storage cost (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/object-lifecycle-mgmt.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Creates a new lifecycle configuration for the bucket or replaces an existing lifecycle configuration. | s3:PutLifecycleConfiguration | | Puts a S3 Intelligent-Tiering configuration to the specified bucket | s3:PutIntelligentTieringConfiguration |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Delete objects by using lifecycle | [Medium (5.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | |

#### Delete objects by using lifecycle

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T25 | | **Name** | Delete objects by using lifecycle | | **Description** | Lifecycle allows you to delete objects after its configured expiry. An attacker can use a lifecycle configuration to destroy data. | | **Goal** | Data manipulation | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Medium (5.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutLifecycleConfiguration"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Identify and ensure the protection of all internal buckets hosting your objects**    Track all buckets you control, define their authorized data classification, identify whether the hosted data is primary (i.e. source of truth, for example logs, backups, forensic data, raw data, etc.) or an input/output of a process (e.g. file-processing, software package, etc.), their WORM requirements (e.g. SEC 17a-4, CTCC, etc.), if they are production/non-production (preferably done at account-level), their storage class. You may use tags, Infra-as-code, AWS Glue Data Catalog, or external management tools like [FINRA herd](https://finraos.github.io/herd/)) | Very High | 1 | - | - |
| **Use S3 Object Lock to protect data integrity**    Implement the authorized default S3 Object Lock on each bucket (note: Amazon S3 evaluates and applies bucket policies before applying bucket default S3 Object Lock settings) | Very High | - | 1 | - |
| **Protect primary data against loss**    Backup primary data in a secure location under a different security authority (e.g. in an [AWS data bunker account](https://wellarchitectedlabs.com/security/100_labs/100_create_a_data_bunker/1_instructions/) via replication, or using AWS Backup for Amazon S3) | High | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Metrics *(subclass of Bucket, FC14)* *You can configure metrics to get additional insights into your usage (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/user-guide/configure-metrics.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Sets or updates a metrics configuration for the CloudWatch request metrics (specified by the metrics configuration ID) from the bucket. | s3:PutMetricsConfiguration |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S3 Storage Lens *(subclass of Bucket, FC31)* *S3 Storage Lens provides a single view of object storage usage and activity across your entire S3 storage.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Puts an Amazon S3 Storage Lens configuration | s3:PutStorageLensConfiguration |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Website *(subclass of Bucket, FC16)* *You can host a static website on Amazon S3. On a static website, individual web pages include static content. They might also contain client-side scripts (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/WebsiteHosting.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Sets the configuration of the website that is specified in the website subresource. | s3:PutBucketWebsite |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Embed client-side script malware in bucket website | [Medium (5.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | Clickjacking on S3 website | [Medium (4.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:L/A:N) | | Read data in transit on the website endpoint | [Low (3.1)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N) | |

#### Embed client-side script malware in bucket website

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T15 | | **Name** | Embed client-side script malware in bucket website | | **Description** | S3 website enables users to be served client-side scripts (e.g. JavaScript). An attacker can upload a client-side script with malware (e.g. cryptomining) to the visitor. | | **Goal** | Direct Financial Gain | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Medium (5.5)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutObject"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Identify and ensure the protection of all external buckets hosting your objects**    Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel).    Scan all data before uploading to an external bucket to ensure the classification of the data is aligned with the bucket classification (e.g. using Macie). | High | 2 | - | - |
| **Scan input/output objects for malware**    If the bucket is used as an input or the output of a process, scan the objects for malware (e.g. using [VirusScan](https://s3-virusscan.widdix.net/), [Cloud Storage Security](https://cloudstoragesec.com/amazon-s3), or [Trend Micro Cloud One](https://aws.amazon.com/blogs/apn/amazon-s3-malware-scanning-using-trend-micro-cloud-one-and-aws-security-hub/)) | Low | - | - | 1 |

#### Clickjacking on S3 website

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T29 | | **Name** | Clickjacking on S3 website | | **Description** | S3 does not enforce certain security headers by default. An attacker can use an iFrame on your website to trick users to interact with their own scripts. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0040](https://attack.mitre.org/tactics/TA0040) | | **CVSS** | [Medium (4.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:L/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Deploy only authorized S3 website and are placed behind a CDN**    Maintain a list of authorized buckets to be configured as a S3 website endpoint    Ensure only authorized buckets are configured as a S3 website endpoint    Ensure S3 website endpoints are protected with HTTP headers ([ref](https://aws.amazon.com/blogs/networking-and-content-delivery/adding-http-security-headers-using-lambdaedge-and-amazon-cloudfront/)) using a CDN (e.g. CloudFront) | Very High | 3 | - | - |

#### Read data in transit on the website endpoint

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T13 | | **Name** | Read data in transit on the website endpoint | | **Description** | S3 website endpoint is serving HTTP only. An attacker can intercept the traffic you send to an external bucket to read the data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Low (3.1)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block direct public access**    Front buckets that are required to be public, using authenticated CDN (e.g. CloudFront) or API Gateway, protected with WAF (e.g. for [hotlinking](https://aws.amazon.com/blogs/security/how-to-prevent-hotlinking-by-using-aws-waf-amazon-cloudfront-and-referer-checking/)) | Very High | 1 | - | - |
| **Deploy only authorized S3 website and are placed behind a CDN**    Maintain a list of authorized buckets to be configured as a S3 website endpoint    Ensure only authorized buckets are configured as a S3 website endpoint    Ensure S3 website endpoints are protected with HTTP headers ([ref](https://aws.amazon.com/blogs/networking-and-content-delivery/adding-http-security-headers-using-lambdaedge-and-amazon-cloudfront/)) using a CDN (e.g. CloudFront) | High | 3 | - | - |
| **Encrypt or tokenize critical data**    Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Medium | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S3 Object Lock *(subclass of Bucket, FC17)* *You can use S3 Object Lock to store objects using a write-once-read-many (WORM) model (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/object-lock-overview.html#object-lock-retention-modes)*). Creating a bucket with S3 Object Lock will enable versioning even without permissions.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Grants permission to allow circumvention of governance-mode object retention settings (for DeleteObject, DeleteObjects and PutObjectRetention) | s3:BypassGovernanceRetention | | Allows placing a default S3 Object Lock configuration at bucket creation (AWS Support needs to be contacted for existing buckets). It automatically enables versioning, even without permission. | s3:PutBucketObjectLockConfiguration | | Puts object retention on a specific object | s3:PutObjectRetention |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Legal hold *(subclass of S3 Object Lock, FC29)* *A legal hold provides the same protection as a retention period, but it has no expiration date. Instead, a legal hold remains in place until you explicitly remove it. Legal holds are independent of retention periods.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Puts Object Lock legal hold on a specific object | s3:PutObjectLegalHold |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Transfer Acceleration *(subclass of Bucket, FC18)* *You can use Transfer Acceleration to improve the performance of long-distance transfers (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/transfer-acceleration.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Sets the Transfer Acceleration state of an existing bucket. | s3:PutAccelerateConfiguration |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Notification *(subclass of Bucket, FC20)* *You can receive notifications when certain events happen in your bucket. Notifications can be sent cross-account (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/NotificationHowTo.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Enables you to receive notifications when certain events happen in your bucket. | s3:PutBucketNotification |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Exfiltrate data via event notification | [Low (2.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:L/I:N/A:N) | |

#### Exfiltrate data via event notification

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T41 | | **Name** | Exfiltrate data via event notification | | **Description** | Event notification sends the key to any configured SQS, SNS or Lambda (cross-account), or EventBridge (same account). An attacker can use the name of objects (1KB) to exfiltrate data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Low (2.4)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:N/S:U/C:L/I:N/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutBucketNotification"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Control event receivers**    Maintain a list of authorized notification receiver(s) (e.g. SNS topic, Lambda, etc.) for each bucket. You might use a simpler approach by using authorized account ID(s) to ensure all your receivers are in authorized AWS account(s).    Ensure only authorized notification receiver(s) (e.g. SNS topic, Lambda, etc.) for each bucket are configured | High | 2 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |
| **Enforce good coding practice**    Do not include sensitive data in bucket names, access point names, object names, object metadata and tags. | Low | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Access point *(subclass of Bucket, FC26)* Access points are named network endpoints that are attached to buckets that you can use to perform S3 object operations. Only certain operations and AWS services are compatible ( Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Creates a new access point. | s3:CreateAccessPoint |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Grant unauthorized access to a bucket by changing/deleting access point policy | [Medium (6.8)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:C/C:N/I:H/A:N) | | Unauthorized collection of data by swapping access point | [Medium (4.6)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:H/PR:H/UI:R/S:U/C:H/I:L/A:N) | |

#### Grant unauthorized access to a bucket by changing/deleting access point policy

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T54 | | **Name** | Grant unauthorized access to a bucket by changing/deleting access point policy | | **Description** | Access point policy can enable access to objects owned by the bucket. An attacker (or someone by negligence) can change the access point policy and make the content accessible. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (6.8)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:C/C:N/I:H/A:N) | | **IAM Access** | {  "OR": ["s3:PutAccessPointPolicy", "s3:DeleteAccessPointPolicy", "s3:PutAccessPointPublicAccessBlock"]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block direct public access**    Enable S3 Block Public Access on all S3 access points (including multi-region), with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Very High | 1 | - | - |
| **Restrict access point access to VPC when in use**    Limit access via the S3 access point by using a VPC endpoint and/or bucket policy with the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" in an allow statement to reduce the length of the allowlist bucket name in the VPC endpoint/bucket policy. | Medium | - | 1 | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

#### Unauthorized collection of data by swapping access point

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T28 | | **Name** | Unauthorized collection of data by swapping access point | | **Description** | Access points can be deleted and recreated with the same name, and therefore the same ARN. An attacker can delete an access point and recreate the same, on a bucket (in the same account) it controls to collect/modify data; or making it accessible over the Internet. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0009](https://attack.mitre.org/tactics/TA0009) | | **CVSS** | [Medium (4.6)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:H/PR:H/UI:R/S:U/C:H/I:L/A:N) | | **IAM Access** | {  "AND": ["s3:CreateAccessPoint", "s3:DeleteAccessPoint"]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Restrict access point access to VPC when in use**    Maintain a list of authorized access between VPCs, S3 access points, and S3.    In the S3 bucket policy, deny all IAM principals not using an authorized S3 access point(s) using the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn"    Block the creation "s3:CreateAccessPoint" of non-VPC S3 access point (e.g. using the condition "StringNotEquals": {"s3:AccessPointNetworkOrigin": "VPC"})    Block all traffic from Internet-configured S3 access point (e.g. on the bucket policy, using a deny statement with the condition "StringNotEquals": {"s3:AccessPointNetworkOrigin": "VPC"}) | Very High | 1 | 3 | - |
| **Block requests with KMS keys from unauthorized AWS account(s)**    Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account    Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account)    Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Very High | 1 | 1 | 1 |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Medium | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S3 Object Lambda *(subclass of Access point, FC32)* *S3 Object Lambda enables users to apply their custom code to process the output of a standard S3 request by automatically invoking a Lambda function.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Creates an Object Lambda access point | s3:CreateAccessPointForObjectLambda | | Grants permission to retrieve objects from Amazon S3 | s3-object-lambda:GetObject | | Grants permission to add an object to a bucket | s3-object-lambda:PutObject |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Hijack connection with an Object Lambda | [Medium (5.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:H/UI:N/S:U/C:H/I:H/A:N) | |

#### Hijack connection with an Object Lambda

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T46 | | **Name** | Hijack connection with an Object Lambda | | **Description** | Object Lambda is invoked between the access point and the object. An attacker can configure a Lambda to modify, snoop, or exfiltrate data. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:H/PR:H/UI:N/S:U/C:H/I:H/A:N) | | **IAM Access** | {  "OR": ["s3:CreateAccessPointForObjectLambda", "s3:PutAccessPointConfigurationForObjectLambda"]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Enforce only authorized Object Lambda access point and associated access**    Maintain a list of authorized Lambda function for each Object Lambda access point, its associated access point, its associated GET request(s), and payload    Ensure only authorized Lambda function for each Object Lambda access point, its associated access point, its associated GET request(s), and payload are created    Ensure Lambda functions configured on Object Lambda access point are secured using Lambda ThreatModel    Maintain a list of cross-account access on each Object Lambda access point    Ensure only authorized cross-account IAM entities are allowed in the Object Lambda access point policy    Ensure CloudWatch is enabled for all Object Lambda access points | Very High | 6 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Multi-Region Access Points *(subclass of Bucket, FC33)* *S3 Multi-Region Access Points provide a single global endpoint to access a data set that spans multiple S3 buckets in different AWS Regions.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Creates a Multi-Region Access Point and associates it with the specified buckets. | s3:CreateMultiRegionAccessPoint |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Grant unauthorized access to buckets by changing the Multi-Region Access Point policy | [Medium (6.8)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:C/C:N/I:H/A:N) | | Gain unauthorized access to buckets trusting all Multi-Region Access Points | [Medium (5.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:R/S:C/C:N/I:H/A:N) | |

#### Grant unauthorized access to buckets by changing the Multi-Region Access Point policy

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T55 | | **Name** | Grant unauthorized access to buckets by changing the Multi-Region Access Point policy | | **Description** | Multi-Region Access Point policy can enable access to objects owned by the bucket. An attacker (or someone by negligence) can change the Multi-Region Access Point policy and make the content accessible. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0008](https://attack.mitre.org/tactics/TA0008) | | **CVSS** | [Medium (6.8)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:C/C:N/I:H/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutMultiRegionAccessPointPolicy"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block direct public access**    Enable S3 Block Public Access on all S3 access points (including multi-region), with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Very High | 1 | - | - |
| **Restrict access point access to VPC when in use**    Limit access via the S3 access point by using a VPC endpoint and/or bucket policy with the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" in an allow statement to reduce the length of the allowlist bucket name in the VPC endpoint/bucket policy.    In the S3 bucket policy, deny all IAM principals not using an authorized S3 access point(s) using the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" | Medium | - | 2 | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions.    In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Medium | 2 | - | - |

#### Gain unauthorized access to buckets trusting all Multi-Region Access Points

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T56 | | **Name** | Gain unauthorized access to buckets trusting all Multi-Region Access Points | | **Description** | Buckets used by Multi-Region Access Points can be configured to delegate their access to any MRAP using the condition "s3:DataAccessPointAccount". An attacker can create a MRAP, add any misconfigured bucket and gain access to it. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (5.7)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:A/AC:L/PR:H/UI:R/S:C/C:N/I:H/A:N) | | **IAM Access** | {  "AND": ["s3:CreateMultiRegionAccessPoint", "s3:PutMultiRegionAccessPointPolicy"]  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Restrict access point access to VPC when in use**    In the S3 bucket policy, deny all IAM principals not using an authorized S3 access point(s) using the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" | Very High | - | 1 | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Medium | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CORS *(subclass of Bucket, FC22)* ***[NOT RECOMMENDED]*** *To configure your bucket to allow cross-origin requests, you create a CORS configuration, which is an XML document with rules that identify the origins that you will allow to access your bucket, the operations (HTTP methods) that will support for each origin, and other operation-specific information. This feature class is NOT RECOMMENDED to be activated since it is all HTTP. Prefer the usage of CDN (e.g. CloudFront), API Gateway, and/or WAF fronting S3 buckets.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Sets the CORS configuration for your bucket. | s3:PutBucketCORS |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bucket default encryption *(subclass of Bucket, FC23)* *You can set default encryption on a bucket so that all new objects are encrypted when stored in the bucket.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Sets the default encryption configuration for the bucket. | s3:PutEncryptionConfiguration |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S3 Object Ownership *(subclass of Bucket, used by Object operations, FC30)* *Enables bucket owners to automatically assume ownership of objects uploaded to their buckets by other AWS accounts. When the object is Put with an ACL of bucket-owner-full-control, the object will be fully owned by the target bucket owner. If the ACL is added later, the ownership is kept by the object owner (*[*ref*](https://docs.aws.amazon.com/AmazonS3/latest/dev/about-object-ownership.html)*).* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Creates or modifies OwnershipControls for an Amazon S3 bucket | s3:PutBucketOwnershipControls |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | None | None | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Public Access Block (bucket) *(subclass of Bucket, FC24)* *S3 Block Public Access (bucket) provides controls at the individual S3 bucket level to ensure objects never have public access.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Creates or modifies the PublicAccessBlock configuration for an Amazon S3 bucket. | s3:PutBucketPublicAccessBlock |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Reduce bucket security by modify the bucket's Public Access Block | [Medium (4.9)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:H/A:N) | |

#### Reduce bucket security by modify the bucket's Public Access Block

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T52 | | **Name** | Reduce bucket security by modify the bucket's Public Access Block | | **Description** | Bucket Public Access Block protects individual buckets from leakage (e.g. object ACL set to public). An attacker can remove this protection by modifying the bucket Public Access Block. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0008](https://attack.mitre.org/tactics/TA0008) | | **CVSS** | [Medium (4.9)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:H/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutBucketPublicAccessBlock"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Disabling ACLs for all buckets**    Ensure bucket ACL and object ACL are disabled on each bucket    Prevent the creation of buckets with ACL enabled (e.g. by using a SCP and/or an IAM policy on "s3:CreateBucket" with a deny statement on StringNotEquals "s3:x-amz-object-ownership":"BucketOwnerEnforced"). Note that it does not block someone from enabling an ACL afterward via PutPutBucketOwnershipControls. | Very High | 1 | 1 | - |
| **Monitor S3 with Amazon GuardDuty and Macie**    Enable and monitor [S3 protection in Amazon GuardDuty](https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_finding-types-s3.html) in all AWS accounts in all Regions, and protect it using GuardDuty ThreatModel. Ensure findings are investigated (e.g. using Amazon Detective). | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Medium | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Public Access Block (account) *(subclass of Bucket, FC25)* *S3 Block Public Access (account) provides controls across an entire AWS account to ensure objects never have public access.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | Creates or modifies the PublicAccessBlock configuration for an AWS account. | s3:PutAccountPublicAccessBlock |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Reduce bucket security by modify the account's Public Access Block | [Medium (4.9)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:H/A:N) | |

#### Reduce bucket security by modify the account's Public Access Block

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T53 | | **Name** | Reduce bucket security by modify the account's Public Access Block | | **Description** | Account Public Access Block protects all buckets of an AWS account from leakage (e.g. object ACL set to public). An attacker can remove this protection by modifying the account's Public Access Block. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0008](https://attack.mitre.org/tactics/TA0008) | | **CVSS** | [Medium (4.9)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:N/I:H/A:N) | | **IAM Access** | {  "UNIQUE": "s3:PutAccountPublicAccessBlock"  } | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Disabling ACLs for all buckets**    Ensure bucket ACL and object ACL are disabled on each bucket    Prevent the creation of buckets with ACL enabled (e.g. by using a SCP and/or an IAM policy on "s3:CreateBucket" with a deny statement on StringNotEquals "s3:x-amz-object-ownership":"BucketOwnerEnforced"). Note that it does not block someone from enabling an ACL afterward via PutPutBucketOwnershipControls. | Very High | 1 | 1 | - |
| **Monitor S3 with Amazon GuardDuty and Macie**    Enable and monitor [S3 protection in Amazon GuardDuty](https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_finding-types-s3.html) in all AWS accounts in all Regions, and protect it using GuardDuty ThreatModel. Ensure findings are investigated (e.g. using Amazon Detective). | Medium | 1 | - | - |
| **Limit the access to the IAM actions required to execute the threats**    Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Medium | 1 | - | - |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Other uses *(class, FC28)* *Others can use their S3 service to impact you in some ways.* Data Flow Diagram (DFD) | Actions and IAM Permissions to deny the feature  |  |  | | --- | --- | | **Action** | **IAM Permission** | | None | None |  Threat List  |  |  | | --- | --- | | **Name** | **CVSS** | | Exfiltrate data by using the public endpoint to upload data in an attacker bucket, using external credentials | [Medium (6.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N) | | Recon on the AWS Region of a bucket | [Medium (4.3)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:L/I:N/A:N) | | Phishing using trademarks | [Low (3.1)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N) | | Uncontrolled change in IAM managed policies | [Low (3.0)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:U/C:L/I:L/A:N) | | Recon of AWS root account emails using the email ACL grantee feature | [Low (2.0)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:P/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N) | | Recon on valid AWS account or IAM principals | [Low (2.0)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:P/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N) | |

#### Exfiltrate data by using the public endpoint to upload data in an attacker bucket, using external credentials

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T10 | | **Name** | Exfiltrate data by using the public endpoint to upload data in an attacker bucket, using external credentials | | **Description** | AWS authenticates per AWS account. An attacker can use their own credentials to exfiltrate data to external S3 buckets through the S3 public endpoint. It can be a non-authenticated user as well. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Medium (6.2)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Block S3 endpoints in your corporate perimeter security**    Block S3 endpoints ([DNS](https://docs.aws.amazon.com/general/latest/gr/s3.html) and [IP ranges](https://aws.amazon.com/premiumsupport/knowledge-center/s3-find-ip-address-ranges/)) in your corporate perimeter security to the Internet (e.g. firewalls, or cloud interception proxy like [Kivera](https://kivera.io)) including via Internet Gateway, to force usage of VPC endpoints. It will block data-plane transfer. Note: AWS console stays functional as it proxies non-data-plane requests (via "console.aws.amazon.com"). | Very High | 1 | - | - |
| **Restrict access point access to VPC when in use**    Maintain a list of authorized access between VPCs, S3 access points, and S3.    Limit access via the S3 access point by using a VPC endpoint and/or bucket policy with the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" in an allow statement to reduce the length of the allowlist bucket name in the VPC endpoint/bucket policy.    Block all traffic from Internet-configured S3 access point (e.g. on the bucket policy, using a deny statement with the condition "StringNotEquals": {"s3:AccessPointNetworkOrigin": "VPC"}) | Very High | 1 | 2 | - |

#### Recon on the AWS Region of a bucket

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T32 | | **Name** | Recon on the AWS Region of a bucket | | **Description** | Error messages can give some information about specific buckets. An attacker who knows the bucket name can find its AWS Region. To find the AWS Region, use "aws s3 presign bucket-name/whatever", go to the presign link, and the error message will give you the region, if not in the right region. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0043](https://attack.mitre.org/tactics/TA0043) | | **CVSS** | [Medium (4.3)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:L/I:N/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| None | | | | |

#### Phishing using trademarks

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T23 | | **Name** | Phishing using trademarks | | **Description** | S3 provides URLs to buckets using the bucket name (i.e. "*mybucket*.s3.amazonaws.com"). An attacker can create a bucket with the name of your trademark to phish users. | | **Goal** | Data theft | | **MITRE ATT&CK®** | [TA0010](https://attack.mitre.org/tactics/TA0010) | | **CVSS** | [Low (3.1)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Protect and/or claim your domains and trademarks/copyrights**    Protect and/or claim your domains and trademarks/copyrights (by creating your trademark buckets and using the [copyright infringement process](https://aws.amazon.com/terms/#notice-and-procedure-for-making-claims-of-copyright-infringement) from AWS) | High | 1 | - | - |

#### Uncontrolled change in IAM managed policies

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T57 | | **Name** | Uncontrolled change in IAM managed policies | | **Description** | AWS managed policies can be attached to your IAM entities, but their permissions are managed by AWS. An attacker (including AWS insider via a [service-linked role](https://aws.amazon.com/security/security-bulletins/AWS-2021-007/)) can use an over-privileged managed permission to execute an attack. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0003](https://attack.mitre.org/tactics/TA0003) | | **CVSS** | [Low (3.0)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:L/AC:H/PR:H/UI:N/S:U/C:L/I:L/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Ensure all requests are blocked from unauthorized service roles**    Ensure all requests are blocked from unauthorized service roles (e.g. by denying all requests with the principal "arn:aws:iam::\*:\*/AWSServiceRoleFor\*" on S3 bucket policies) | Very High | 1 | - | - |

#### Recon of AWS root account emails using the email ACL grantee feature

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T19 | | **Name** | Recon of AWS root account emails using the email ACL grantee feature | | **Description** | S3 allows you to add root account emails in ACL ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#specifying-grantee)), and as well resolve the given canonical ID into an AWS account ID (via a bucket policy, which automatically resolves a canonical ID into an ARN). An attacker can do trial-and-error to discover existing AWS root account emails and related AWS account ID (even if you do not use the region where the feature is available); and use this information to launch another attack (e.g. phishing). | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0043](https://attack.mitre.org/tactics/TA0043) | | **CVSS** | [Low (2.0)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:P/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Use an unguessable naming convention**    Use an unguessable naming convention for the email addresses of your AWS accounts (e.g. add a + sign and a random string to redirect the email in the same mailbox) | High | 1 | - | - |

#### Recon on valid AWS account or IAM principals

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Threat Id** | S3.T24 | | **Name** | Recon on valid AWS account or IAM principals | | **Description** | AWS provides error messages in the S3 bucket policy that can be used for basic recon. An attacker can discover whether an AWS account with a specific AWS account ID or AWS IAM principal exists by modifying the S3 policy to grant some rights to the said AWS account/IAM principal. | | **Goal** | Launch another attack | | **MITRE ATT&CK®** | [TA0043](https://attack.mitre.org/tactics/TA0043) | | **CVSS** | [Low (2.0)](https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:P/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N) | | **IAM Access** | {} | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control Objectives** | **Priority** | **# of associated Controls** | | |
| **Directive** | **Preventative** | **Detective** |
| **Use an unguessable naming convention**    Use an unguessable naming convention for your IAM users and IAM roles (e.g. add a random string) | High | 1 | - | - |

# Control Implementation

## Enforce encryption-in-transit [S3.CO1]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C1, depends on S3.C119, assured by S3.C2]  Block all unencrypted requests and unauthorized TLS version(s) from IAM entities you control (e.g. by denying all unencrypted requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" !=*authorized TLS version(s)*, using an SCP on your AWS Organization root node) | Make an unencrypted S3 API call, it should be denied. | Low | S3.FC1  S3.FC5 | S3.T12 (Very High)  S3.T34 (High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C2]  Verify the control blocking unencrypted requests and unauthorized TLS version(s) from IAM entities you control (e.g. an SCP on your AWS Organizations root node) is properly implemented | Remove the control blocking unencrypted requests and unauthorized TLS version(s) (e.g. the SCP on your root node), it should be detected. | High | S3.FC1  S3.FC5 | - | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C3, depends on S3.C119, assured by S3.C5]  Block all unencrypted requests and unauthorized TLS version(s) from VPC endpoints you control (e.g. by denying all requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" != *authorized TLS version(s)*, on the VPC endpoint policy) | Make an unencrypted AWS API call from one of your VPCs with VPC endpoint, it should be denied. | Low | S3.FC1  S3.FC5 | S3.T12 (Medium)  S3.T34 (Medium) | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C4]  Monitor and investigate that all requests made with HTTP (e.g., via CloudTrail S3 data events with the lack of additionalEventData.CipherSuite) | Make an unencrypted AWS API call from one of your VPCs with VPC endpoint, it should be detected. | Low | S3.FC1  S3.FC5 | S3.T12 (Low)  S3.T34 (Low) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C5]  Verify a statement exists on all your VPC endpoint policy denying all requests with the condition "aws:SecureTransport" = False | Create/remove the statement on a VPC endpoint policy denying 1) all unencrypted requests or 2) unauthorized TLS version(s), it should be detected. | High | S3.FC1  S3.FC5 | - | Medium |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C6, depends on S3.C119, assured by S3.C7]  Block all unencrypted requests to S3 bucket you control (e.g. by denying all requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" != *authorized TLS version(s)*, on the S3 bucket policy) | Make an unencrypted AWS API call to a bucket you control, it should be denied. | Low | S3.FC5 | S3.T34 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C7]  Verify all S3 bucket policies block unencrypted traffic (e.g. using the AWS Config rule: [S3\_BUCKET\_SSL\_REQUESTS\_ONLY](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-ssl-requests-only.html)) and unauthorized version(s) of TLS. | Remove the statement on a S3 bucket policy 1) denying all unencrypted requests and 2) denying unauthorized TLS versions, it should be detected. | Medium | S3.FC5 | - | High |
| Directive (COSO)  Identify (NIST CSF) | [S3.C119]  Maintain a list of authorized version(s) of TLS/SSL per bucket (or per account/OU/Org) | Request the list of authorized version(s) of TLS/SSL per bucket (or per account/OU/Org), its review mechanism and associated records | Low | S3.FC1 | S3.T12 (Very Low) | High |

## Block S3 endpoints in your corporate perimeter security [S3.CO2]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C8]  Block S3 endpoints ([DNS](https://docs.aws.amazon.com/general/latest/gr/s3.html) and [IP ranges](https://aws.amazon.com/premiumsupport/knowledge-center/s3-find-ip-address-ranges/)) in your corporate perimeter security to the Internet (e.g. firewalls, or cloud interception proxy like [Kivera](https://kivera.io)) including via Internet Gateway, to force usage of VPC endpoints. It will block data-plane transfer. Note: AWS console stays functional as it proxies non-data-plane requests (via "console.aws.amazon.com"). | Request the evidence of the implementation of blocking S3 endpoints in your corporate perimeter security (e.g. firewalls) and tests of its effectiveness. | Low | S3.FC1  S3.FC28  S3.FC5  S3.FC7 | S3.T7 (High)  S3.T10 (High)  S3.T12 (Low)  S3.T18 (Medium)  S3.T34 (Very High) | High |

## Enable CloudTrail S3 data events [S3.CO3]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Detect (NIST CSF) | [S3.C9]  Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Request the CloudTrail ThreatModel and the evidence of its application for enabling and protecting S3 data events | Very Low | S3.FC1  S3.FC5  S3.FC8 | S3.T1 (Low)  S3.T4 (Low)  S3.T5 (Low)  S3.T6 (Low)  S3.T7 (Low)  S3.T8 (Low)  S3.T9 (Low)  S3.T11 (Low)  S3.T12 (Low)  S3.T16 (Low)  S3.T21 (Low)  S3.T31 (Low)  S3.T34 (Low)  S3.T35 (Low)  S3.T36 (Low)  S3.T39 (Low) | Medium |

## Monitor S3 with Amazon GuardDuty and Macie [S3.CO4]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Detect (NIST CSF) | [S3.C10]  Enable and monitor [S3 protection in Amazon GuardDuty](https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_finding-types-s3.html) in all AWS accounts in all Regions, and protect it using GuardDuty ThreatModel. Ensure findings are investigated (e.g. using Amazon Detective). | Request the GuardDuty ThreatModel and the evidence of its application for enabling, monitoring, investigation and protecting S3 protection | Low | S3.FC1  S3.FC24  S3.FC25  S3.FC5  S3.FC8 | S3.T3 (Low)  S3.T4 (Low)  S3.T16 (Medium)  S3.T52 (Medium)  S3.T53 (Medium) | Medium |
| Directive (COSO)  Detect (NIST CSF) | [S3.C118]  Enable [S3 policy findings in Amazon Macie](https://docs.aws.amazon.com/macie/latest/user/findings-types.html#findings-policy-types) in all AWS accounts in all Regions, and protect it using Macie ThreatModel | Request the Macie ThreatModel and the evidence of its application for enabling and protecting S3 policy findings | Very Low | S3.FC10  S3.FC15  S3.FC5  S3.FC8 | S3.T2 (Medium)  S3.T4 (Medium)  S3.T22 (Medium)  S3.T36 (Medium)  S3.T37 (Medium)  S3.T38 (Medium) | High |

## Identify and ensure the protection of all external buckets hosting your objects [S3.CO5]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C11]  Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel). | Request the list of all authorized external buckets authorized to host your objects, their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), their data classification and the mechanism used to ensure the security of those buckets | Medium | S3.FC1  S3.FC16  S3.FC5 | S3.T3 (High)  S3.T5 (Very Low)  S3.T6 (Low)  S3.T7 (Very Low)  S3.T8 (Very Low)  S3.T9 (Very Low)  S3.T11 (Low)  S3.T14 (Very Low)  S3.T15 (Very Low)  S3.T21 (Very Low)  S3.T31 (High)  S3.T43 (Very High) | Very High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C12, depends on S3.C11]  Allow only authorized ACL on objects for buckets you don't control (e.g. using IAM and VPC endpoint policy with the [ACL conditions](https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#acl-specific-condition-keys)) | Put an object with an unauthorized ACL, it should be denied. | Medium | S3.FC1 | S3.T5 (Medium)  S3.T6 (High) | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C13, depends on S3.C11]  Monitor that only authorized external buckets are used (e.g. via CloudTrail S3 data events in resources[].accountId and resources[].ARN). Both account ID and bucket name must be verified. | Make a call to an unauthorized bucket, it should be detected | Low | S3.FC1  S3.FC5 | S3.T1 (Low)  S3.T7 (Low)  S3.T11 (Low)  S3.T21 (Low)  S3.T31 (Medium) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C14, depends on S3.C11]  Scan all data before uploading to an external bucket to ensure the classification of the data is aligned with the bucket classification (e.g. using Macie). | Request 1) the mechanism ensuring all data are scanned for proper data classification before upload to an external bucket are configured, 2) its records of execution for all object upload flows, and 3) plan to move any older object upload flows | High | S3.FC1  S3.FC16  S3.FC5 | S3.T5 (High)  S3.T14 (High)  S3.T15 (Medium) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C15]  Request access via an S3 access point on a bucket you don't own, if compatible with your interaction with the bucket (e.g. not through not-compatible AWS service) | Request the documented reason(s) access point was not implemented in the use case | Low | S3.FC1 | S3.T8 (Medium)  S3.T9 (Medium)  S3.T31 (Very High) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C114, depends on S3.C11]  For all external buckets with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover, block the PutObject with any ACL (e.g. using IAM or SCP and a deny on the condition "StringLike": {"s3:x-amz-acl": "\*"}). It should be called via PutObjectAcl. | Make a request to an external bucket with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover requirement, it should be denied. | High | S3.FC1 | S3.T43 (Very High) | High |
| Detective (COSO)  Detect (NIST CSF) | [S3.C115, depends on S3.C11]  For all external bucket with bucket-owner-full-control ACL requirements but without S3 Object Ownership handover, monitor that the PutObject do not include the ACL operation | Make a request to an external bucket with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover requirement, it should be detected. | Low | S3.FC1 | S3.T43 (Low) | Medium |

## Model the threats on all AWS services accessing S3 [S3.CO6]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C16]  Analyze and protect all AWS services accessing S3 (e.g. via ThreatModel). Enforce usage in VPC only, whenever possible. | Request the threat and mitigating controls for all AWS services using S3. | High | S3.FC1  S3.FC5 | S3.T21 (Very High)  S3.T30 (Very High) | Medium |

## Limit and monitor access via S3 VPC endpoints [S3.CO7]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C17]  For each VPC, maintain a list of AWS Organizations, OU, and/or AWS account(s) where IAM entities are authorized to access S3 | For each VPC, request the list of AWS Organizations, OU, and/or AWS account(s), where IAM entities are authorized to access S3, its review process, and its review records | Medium | S3.FC1  S3.FC5 | S3.T9 (Very Low)  S3.T11 (Very Low) | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C18, depends on S3.C17]  For each VPC with an IAM entity allowed to use S3, secure them with the VPC ThreatModel (e.g. [modification of VPC endpoints, VPC endpoint policy, routing table, Security Groups](https://docs.aws.amazon.com/vpc/latest/userguide/vpce-gateway.html)) | Request how VPC ThreatModel for S3 is being applied. | High | S3.FC1  S3.FC5 | S3.T9 (Medium)  S3.T11 (Medium) | Low |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C19, depends on S3.C17, assured by S3.C20]  Block any IAM entity not belonging to an authorized AWS Organizations, OU, and/or AWS account(s) to call S3 from your VPCs by adding a deny statement on the S3 VPC endpoint policy of each VPC, with the condition using "aws:PrincipalOrgPaths" ([ref](https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_condition-keys.html#condition-keys-principalorgpaths)) including the full Org ID, as those are globally unique. | For each VPC, do an API call with an IAM entity which is not part of its authorized AWS Organizations path(s), it should be denied. | Low | S3.FC1  S3.FC5 | S3.T9 (Very High)  S3.T11 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C20]  Verify all S3 VPC endpoint are blocking any IAM entity not belonging to an authorized AWS Organizations, OU and/or AWS account(s) | Remove the policy statement blocking any IAM entity not belonging to an authorized AWS Organizations, OU and/or AWS account(s) from the VPC endpoint, it should be detected. | High | S3.FC1  S3.FC5 | - | High |
| Directive (COSO)  Detect (NIST CSF) | [S3.C21]  Enable [VPC DNS query logging](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resolver-query-logs.html) in all VPC | Request the mechanism to enable VPC DNS query logging in all VPC | Medium | S3.FC1  S3.FC5 | S3.T8 (Very Low)  S3.T9 (Very Low)  S3.T11 (Very Low) | Very Low |
| Directive (COSO)  Identify (NIST CSF) | [S3.C22]  Maintain a list of authorized S3 and S3 access point (and their respective AWS accounts) to be accessed for each VPC | Request the list of authorized S3 and S3 access point to be access for each VPC, its review process, and its review records | Medium | S3.FC1  S3.FC5 | S3.T8 (Very Low)  S3.T9 (Very Low)  S3.T11 (Very Low) | Medium |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C23, depends on S3.C22, assured by S3.C24]  Limit the access to only authorized S3 bucket(s) or their AWS account(s) from each VPC (e.g. using the condition key "s3:ResourceAccount" on the VPC endpoint policy, alternatively use a specific resource-level statement for each bucket, or if the VPC endpoint policy size is beyond the limit and more granular control on VPC is required, use access points) | Make a request to an unauthorized bucket from one of your VPC, it should be denied | Medium | S3.FC1  S3.FC5 | S3.T8 (High)  S3.T9 (High)  S3.T11 (High) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C24]  Verify all VPC are limited to limit access to only authorized S3 bucket(s) | Remove the control limiting access to only authorized S3 bucket(s), it should be detected. | High | S3.FC1  S3.FC5 | - | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C25, depends on S3.C21,S3.C22]  Monitor VPC DNS query logs that only authorized S3 bucket and S3 access points are being queried in each VPC (e.g. using VPC DNS query logging), and protect it using Route53 ThreatModel | Make a DNS query to an unauthorized 1) S3 bucket and 2) S3 access points, it should be detected. | Low | S3.FC1  S3.FC5 | S3.T8 (Low)  S3.T9 (Low)  S3.T11 (Low) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C124]  Ensure all S3 VPC endpoints (Interface and Gateway) are covered by the VPC endpoints controls | Request the mechanism ensuring all S3 VPC endpoints (Interface and Gateway) are covered by the VPC endpoints controls, and its records | Low | S3.FC1 | S3.T45 (Very High) | Low |

## Limit the access to the IAM actions required to execute the threats [S3.CO8]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C26]  Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Request the list of authorized IAM principals that have the permissions required to execute the threat actions, its review process, and its review records | High | S3.FC1  S3.FC10  S3.FC12  S3.FC13  S3.FC15  S3.FC19  S3.FC2  S3.FC20  S3.FC24  S3.FC25  S3.FC26  S3.FC27  S3.FC32  S3.FC33  S3.FC5  S3.FC6  S3.FC7  S3.FC8 | S3.T1 (High)  S3.T2 (High)  S3.T5 (Low)  S3.T6 (Medium)  S3.T7 (High)  S3.T8 (High)  S3.T11 (High)  S3.T14 (Very High)  S3.T16 (High)  S3.T17 (Very High)  S3.T18 (High)  S3.T21 (Medium)  S3.T25 (High)  S3.T26 (High)  S3.T28 (High)  S3.T30 (High)  S3.T33 (Very High)  S3.T35 (Very High)  S3.T36 (Medium)  S3.T37 (Very High)  S3.T38 (Very High)  S3.T39 (High)  S3.T41 (High)  S3.T42 (High)  S3.T44 (High)  S3.T46 (High)  S3.T47 (High)  S3.T48 (High)  S3.T49 (High)  S3.T50 (High)  S3.T51 (High)  S3.T52 (High)  S3.T53 (High)  S3.T54 (High)  S3.T55 (High)  S3.T56 (High)  S3.T58 (High)  S3.T59 (High) | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C27, assured by S3.C28]  In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Request all S3 bucket/access point/Object Lambda access point policy statements with "allow", no principal from the same account should be authorized. | Low | S3.FC1  S3.FC10  S3.FC12  S3.FC13  S3.FC15  S3.FC19  S3.FC2  S3.FC20  S3.FC26  S3.FC27  S3.FC32  S3.FC33  S3.FC5  S3.FC7 | S3.T1 (Low)  S3.T2 (Low)  S3.T6 (Low)  S3.T7 (Low)  S3.T8 (Low)  S3.T11 (Low)  S3.T14 (Medium)  S3.T16 (Low)  S3.T17 (Medium)  S3.T18 (Low)  S3.T21 (Low)  S3.T25 (Low)  S3.T26 (Medium)  S3.T30 (Low)  S3.T33 (Medium)  S3.T35 (Medium)  S3.T36 (Low)  S3.T37 (Medium)  S3.T38 (Medium)  S3.T39 (Low)  S3.T41 (Low)  S3.T42 (Low)  S3.T44 (Low)  S3.T46 (Medium)  S3.T54 (Medium)  S3.T55 (Medium)  S3.T58 (Medium)  S3.T59 (Medium) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C28]  Verify all S3 bucket/access point/Object Lambda access point policies do not allow an IAM principal of the same AWS account (e.g. using the Config rule S3\_BUCKET\_POLICY\_GRANTEE\_CHECK for bucket policy) | Add an allow statement for an IAM principal of the same account in 1) a bucket policy, 2) in an access point policy, and 3) in an Object Lambda access point, it should be detected. | Medium | S3.FC1  S3.FC10  S3.FC12  S3.FC13  S3.FC15  S3.FC19  S3.FC2  S3.FC20  S3.FC26  S3.FC27  S3.FC32  S3.FC33  S3.FC5  S3.FC7 | - | Medium |
| Directive (COSO)  Identify (NIST CSF) | [S3.C149]  For each bucket, maintain a list of authorized IAM principals allowed to access via bucket policy | Request the list of authorized a list of authorized IAM principals allowed to access via bucket policy, its review process, and its review records | Medium | S3.FC10 | S3.T37 (Very Low) | Very High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C150, depends on S3.C149, assured by S3.C151]  Ensure only authorized a list of authorized IAM principals allowed to access via bucket policy are configured (e.g. using IAM Access Analyzer for the reconciliation) | Request 1) the mechanism ensuring only authorized IAM principals allowed to access via bucket policy are configured, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Medium | S3.FC10 | S3.T37 (Very High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C151]  Verify only authorized IAM principals allowed to access via bucket policy are used (e.g. using the AWS Config rule [S3\_BUCKET\_POLICY\_GRANTEE\_CHECK](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-policy-grantee-check.html)) | Allow an unauthorized IAM principal on a bucket policy, it should be detected. | Medium | S3.FC10 | - | Very High |

## Block requests with KMS keys from unauthorized AWS account(s) [S3.CO9]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C31]  Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account | Request the list of authorized AWS accounts to provide KMS keys for S3 for each AWS account, its review process, and its review records | Medium | S3.FC1  S3.FC15  S3.FC26  S3.FC5  S3.FC8 | S3.T1 (Very Low)  S3.T2 (Very Low)  S3.T4 (Very Low)  S3.T5 (Very Low)  S3.T7 (Very Low)  S3.T8 (Very Low)  S3.T9 (Very Low)  S3.T11 (Very Low)  S3.T16 (Very Low)  S3.T21 (Very Low)  S3.T27 (Very Low)  S3.T28 (Very Low)  S3.T30 (Very Low)  S3.T31 (Very Low) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C32, depends on S3.C31]  Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account) | Make a request encrypted with a KMS key from unauthorized AWS account, it should be denied | Low | S3.FC1  S3.FC15  S3.FC26  S3.FC5  S3.FC8 | S3.T1 (High)  S3.T2 (Medium)  S3.T4 (High)  S3.T5 (High)  S3.T7 (High)  S3.T8 (High)  S3.T9 (High)  S3.T11 (Medium)  S3.T16 (High)  S3.T21 (Medium)  S3.T27 (Low)  S3.T28 (High)  S3.T30 (High)  S3.T31 (High) | High |
| Detective (COSO)  Detect (NIST CSF) | [S3.C33, depends on S3.C31]  Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Make a call to an unauthorized bucket, it should be detected | Low | S3.FC1  S3.FC15  S3.FC26  S3.FC5  S3.FC8 | S3.T1 (Low)  S3.T2 (Low)  S3.T4 (Low)  S3.T5 (Low)  S3.T7 (Low)  S3.T8 (Low)  S3.T9 (Low)  S3.T11 (Low)  S3.T16 (Low)  S3.T21 (Low)  S3.T27 (Very Low)  S3.T28 (Low)  S3.T30 (Low)  S3.T31 (Low) | Low |

## Block changes to make an object public via object ACL [S3.CO10]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C34, assured by S3.C36]  Deny requests to change object ACL to public (e.g. using an SCP, S3 bucket policy, and VPC endpoint policy blocking PutObjectAcl for "s3:x-amz-grant-read", "s3:x-amz-grant-read-acp", "s3:x-amz-grant-write-acp", "s3:x-amz-grant-full-control" on the following predefined groups "http://acs.amazonaws.com/groups/global/AllUsers" and "http://acs.amazonaws.com/groups/global/AuthenticatedUsers") | Make a call to create a public ObjectACL, it should be denied. | Medium | S3.FC1  S3.FC5 | S3.T6 (Very High)  S3.T36 (Very High) | High |
| Detective (COSO)  Detect (NIST CSF) | [S3.C35]  Monitor ObjectACL changed (or tentatively changed) to public using CloudTrail S3 data events | Make a call to create a public ObjectACL, it should be detected. | Low | S3.FC1  S3.FC5 | S3.T6 (Low)  S3.T36 (Low) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C36]  Verify the control blocking change ObjectACL to public (e.g. an SCP and VPC endpoint policy) is properly implemented | Remove the control blocking changes of ObjectACL to public, it should be detected. | High | S3.FC1  S3.FC5 | - | High |
| Detective (COSO)  Detect (NIST CSF) | [S3.C37]  Monitor and investigate anonymous requests to objects (e.g. using CloudTrail S3 data events with userIdentity.accountId=ANONYMOUS\_PRINCIPAL) | Make an anonymous call, it should be detected. | Low | S3.FC5 | S3.T36 (Low) | Low |

## Prevent deletion of buckets [S3.CO11]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C38, assured by S3.C39]  Block the action "s3:DeleteBucket" (e.g. via SCP, exemption can be managed by authorizing a SuperAdmin to delete buckets with a certain tag, and with bucket owners able to tag bucket) | Do a DeleteBucket, it should be denied | Low | S3.FC5 | S3.T1 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C39]  Verify the control blocking the action "s3:DeleteBucket" (e.g. an SCP on your AWS Organizations root node) is properly implemented | Remove the control blocking the action "s3:DeleteBucket" (e.g. an SCP on your root node), it should be detected. | High | S3.FC5 | - | High |
| Detective (COSO)  Detect (NIST CSF) | [S3.C40]  Scan your CNAME records (e.g. in Route53) and CloudFront origin for deleted buckets | Create a CNAME record and CloudFront origin with an invalid bucket, it should be detected. | High | S3.FC5 | S3.T1 (Very Low) | Very Low |

## Enforce good coding practice [S3.CO12]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C41]  Parameterize the S3 bucket name or S3 access point in your code (no hardcoding) | Request the process on ensuring S3 bucket name or S3 access point are not hard-coded | Medium | S3.FC5 | S3.T1 (Low) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C42]  When connecting to S3 endpoints, use virtual-hosted model ("my-bucket-name.s3.amazonaws.com" or "my-bucket-name.my-s3-regional-endpoint.amazonaws.com") instead of path-style model ("s3.amazonaws.com/my-bucket-name" or "my-s3-regional-endpoint.amazonaws.com/my-bucket-name") (see [ref](https://aws.amazon.com/blogs/aws/amazon-s3-path-deprecation-plan-the-rest-of-the-story/)). All the latest SDKs make use of domain style by default. | Request the mechanism ensuring the usage of domain style instead of path style. | Very Low | S3.FC1 | S3.T35 (High) | Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C43]  Monitor that all S3 connections are made with the virtual-hosted model (e.g. via CloudTrail S3 requestParameters.Host) | Make a path-style request to S3, it should be detected. | Medium | S3.FC1 | S3.T35 (Low) | Very Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C44]  Use "x-amz-checksum" from the [object metadata](https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingMetadata.html#SysMetadata) to validate the integrity of the object instead of etag. If etag is used, make sure properly account for its different definitions ([ref](https://teppen.io/2018/06/23/aws_s3_etags/)) | Request 1) the mechanism ensuring checksum are being used instead of etag, and otherwise ensuring etag different definitions are properly accounted for, and 2) plan to move any older system using etag to use the checksum metadata. | Low | S3.FC1  S3.FC5 | S3.T17 (Medium)  S3.T27 (High) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C45]  Do not include sensitive data in bucket names, access point names, object names, object metadata and tags. | Request the process ensuring no sensitive data is included in bucket names, object names, object metadata and tags. | Low | S3.FC12  S3.FC20 | S3.T41 (Low)  S3.T42 (Medium) | Very Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C46]  Ensure all S3 buckets interacted with are in the correct AWS account (e.g. using the condition in all compatible S3 requests: x-amz-expected-bucket-owner and x-amz-source-expected-bucket-owner) | Request the process on ensuring that all S3 buckets interacted with are in the correct AWS account | Medium | S3.FC1  S3.FC5 | S3.T1 (Medium)  S3.T3 (Medium) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C113, depends on S3.C11]  When transmitting an object to an external bucket with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover, use 2 separate APIs (PutObject and PutObjectAcl) instead of the built-in object ACL operation in PutObject. | Request the process on ensuring that PutObject requests on external bucket with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover use 2 separate APIs | Medium | S3.FC1 | S3.T43 (High) | High |

## Block direct public access [S3.CO13]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C47, assured by S3.C48]  Front buckets that are required to be public, using authenticated CDN (e.g. CloudFront) or API Gateway, protected with WAF (e.g. for [hotlinking](https://aws.amazon.com/blogs/security/how-to-prevent-hotlinking-by-using-aws-waf-amazon-cloudfront-and-referer-checking/)) | Request the process ensuring that buckets required to be public are front by authenticated CDN or API Gateway | Medium | S3.FC16  S3.FC5 | S3.T13 (Very High)  S3.T14 (Medium)  S3.T22 (Very High) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C48]  Verify no bucket is available publicly for write or read (e.g. using the AWS Config rules: [S3\_BUCKET\_PUBLIC\_READ\_PROHIBITED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-public-read-prohibited.html) and [S3\_BUCKET\_PUBLIC\_WRITE\_PROHIBITED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-public-write-prohibited.html)) | Create a public S3 bucket, it should be detected. | Very Low | S3.FC16  S3.FC5 | - | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C49, assured by S3.C50]  Enable account-level S3 Block Public Access on all AWS accounts, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Request 1) the mechanism ensuring account-level S3 Block Public Access is enabled on all AWS accounts, 2) its records of execution for all new AWS accounts, and 3) plan to move any older AWS accounts | Very Low | S3.FC10  S3.FC5  S3.FC8 | S3.T4 (High)  S3.T14 (High)  S3.T36 (Very High)  S3.T37 (Very High)  S3.T38 (Medium) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C50]  Verify account-level S3 Block Public Access is enabled on all AWS accounts, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true (e.g. using the AWS Config rule: [S3\_ACCOUNT\_LEVEL\_PUBLIC\_ACCESS\_BLOCKS](https://docs.aws.amazon.com/config/latest/developerguide/s3-account-level-public-access-blocks.html)). | Remove the account-level S3 Block Public Access of an AWS account, it should be detected. | Very Low | S3.FC10  S3.FC5  S3.FC8 | - | Very High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C51, assured by S3.C52]  Enable S3 Block Public Access on all S3 buckets, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Request 1) the mechanism ensuring S3 Block Public Access is enabled on each bucket, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Low | S3.FC10  S3.FC5  S3.FC8 | S3.T4 (High)  S3.T14 (High)  S3.T36 (Very High)  S3.T37 (Very High)  S3.T38 (Medium) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C52]  Verify S3 Block Public Access is enabled on all S3 buckets, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true (e.g. using the AWS Config rule: [S3\_BUCKET\_LEVEL\_PUBLIC\_ACCESS\_PROHIBITED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-level-public-access-prohibited.html)). | Remove a S3 Block Public Access of an S3 bucket, it should be detected. | Very Low | S3.FC10  S3.FC5  S3.FC8 | - | Very High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C53, assured by S3.C54]  Enable S3 Block Public Access on all S3 access points (including multi-region), with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Request 1) the mechanism ensuring S3 Block Public Access is enabled on each S3 access point, 2) its records of execution for all new S3 access points, and 3) plan to move any older S3 access points | Low | S3.FC10  S3.FC26  S3.FC33  S3.FC5 | S3.T14 (High)  S3.T36 (Medium)  S3.T37 (Medium)  S3.T38 (Medium)  S3.T54 (High)  S3.T55 (High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C54]  Verify S3 Block Public Access is enabled on all S3 access points (including multi-region), with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Remove S3 Block Public Access of 1) an access point, and 2) a Multi-Region Access Point, it should be detected | Medium | S3.FC10  S3.FC26  S3.FC33  S3.FC5 | - | Very High |

## Block bucket ACL [S3.CO14]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C55, assured by S3.C57]  Deny requests to add bucket ACL (e.g. using an SCP, bucket policy, and VPC endpoint policy blocking "s3:PutBucketAcl") | Make a call to create a bucket ACL, it should be denied. | Medium | S3.FC19  S3.FC8 | S3.T4 (Very High)  S3.T58 (Very High) | High |
| Detective (COSO)  Detect (NIST CSF) | [S3.C56]  Monitor changes on bucket ACL to ensure it stays private (e.g. using CloudTrail event PutBucketAcl and its fields requestParameters.x-amz-acl should be either "private" or not existing) | Make a call to have a bucket ACL other than private, it should be detected. | Medium | S3.FC19  S3.FC8 | S3.T4 (Low)  S3.T58 (Low) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C57]  Verify the control blocking bucket ACL changes (e.g. an SCP, a bucket policy and VPC endpoint policy) is properly implemented | Remove the control blocking bucket ACL changes, it should be detected. | High | S3.FC19  S3.FC8 | - | High |

## Identify and ensure the protection of all internal buckets hosting your objects [S3.CO15]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C58]  Track all buckets you control, define their authorized data classification, identify whether the hosted data is primary (i.e. source of truth, for example logs, backups, forensic data, raw data, etc.) or an input/output of a process (e.g. file-processing, software package, etc.), their WORM requirements (e.g. SEC 17a-4, CTCC, etc.), if they are production/non-production (preferably done at account-level), their storage class. You may use tags, Infra-as-code, AWS Glue Data Catalog, or external management tools like [FINRA herd](https://finraos.github.io/herd/)) | Request the list of all buckets you control define their authorized data classification, and identify whether the data is primary and the mechanism and records to ensure the accuracy of those metadata | High | S3.FC1  S3.FC13  S3.FC5 | S3.T11 (High)  S3.T17 (Very High)  S3.T25 (Low) | High |
| Detective (COSO)  Detect (NIST CSF) | [S3.C59, depends on S3.C58]  Use a data discovery tool (e.g. Amazon Macie) to ensure no sensitive data is stored in an unauthorized bucket | Upload a higher classification data in a bucket, it should be detected. | Medium | S3.FC5 | S3.T11 (Medium) | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C60]  Use a data discovery tool (e.g. Amazon Macie) to ensure the bucket names, object names, tags, and metadata do not contain sensitive data | Create a bucket name, object name, tags, or a metadata of an object with sensitive data, it should be detected. | Very High | S3.FC5 | S3.T11 (Very Low) | Very Low |

## Enforce encryption-at-rest [S3.CO16]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C61]  Maintain a list of authorized KMS key(s) for each bucket and their default encryption key. You might simplify by using only 1 key per bucket, ideally dedicated. Note that an S3 server access log bucket does not support KMS encryption ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/bucket-encryption.html#bucket-encryption-how-to-set-up)). | Request the list of authorized KMS key(s) for each bucket, its review process, and its review records | Medium | S3.FC1  S3.FC10  S3.FC5 | S3.T11 (Very Low)  S3.T16 (Very Low)  S3.T17 (Very Low)  S3.T20 (Very Low)  S3.T30 (Very Low)  S3.T36 (Very Low)  S3.T37 (Very Low) | Very High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C140, assured by S3.C62]  Ensure all objects on S3 buckets are encrypted with an authorized KMS key | Request the mechanism (including training, or utility) ensuring only authorized KMS key are used for any objects stored in S3 | Medium | S3.FC1  S3.FC10  S3.FC5 | S3.T11 (Medium)  S3.T16 (Medium)  S3.T17 (Medium)  S3.T20 (Medium)  S3.T30 (Medium)  S3.T36 (Medium)  S3.T37 (Medium) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C62]  Verify all objects on S3 buckets are encrypted with an authorized KMS key (e.g. using S3 inventory, see [blog](https://aws.amazon.com/blogs/storage/encrypting-objects-with-amazon-s3-batch-operations/), or [S3 Storage Lens](https://docs.aws.amazon.com/AmazonS3/latest/dev/storage_lens_basics_metrics_recommendations.html#storage_lens_basics_metrics_types)) | Upload an encrypted data using an unauthorized KMS key, it should be detected. | Medium | S3.FC1  S3.FC10  S3.FC5 | - | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C63, depends on S3.C61]  Use KMS ThreatModel to protect the KMS keys used for S3 (e.g. using encryptionContext on the policy of each KMS key) | Request the KMS ThreatModel and the evidence of its application to protect S3 | High | S3.FC1  S3.FC10  S3.FC5 | S3.T17 (Medium)  S3.T36 (Low)  S3.T37 (Low) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C64, depends on S3.C61, assured by S3.C65]  Implement an authorized default encryption key on each bucket; and enable S3 Bucket Key, if CloudTrail events are not required for KMS encrypt/decrypt (note: Amazon S3 evaluates and applies bucket policies before applying bucket default encryption settings) | Request 1) the mechanism implementing an authorized default encryption key on each bucket and enabling S3 Bucket Key, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Low | S3.FC1  S3.FC10  S3.FC5 | S3.T17 (Medium)  S3.T20 (High)  S3.T36 (High)  S3.T37 (High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C65]  Verify each bucket has an authorized default encryption key and has S3 Bucket Key enabled | Create/modify a bucket 1) without default encryption, 2) with a wrong default encryption key or 3) without S3 Bucket Key disabled, it should be detected. | Medium | S3.FC1  S3.FC10  S3.FC5 | - | Very High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C66, depends on S3.C61, assured by S3.C67]  Block PutObject requests with unauthorized KMS key on each bucket (e.g. using an S3 bucket policy deny statement on PutObject if the condition if exists "s3:x-amz-server-side-encryption-aws-kms-key-id" is not an authorized KMS key) | Make a request encrypted with an unauthorized KMS key, it should be denied | Low | S3.FC1  S3.FC10  S3.FC5 | S3.T11 (High)  S3.T16 (High)  S3.T17 (High)  S3.T20 (Very High)  S3.T30 (High)  S3.T36 (High)  S3.T37 (High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C67]  Verify all buckets block PutObject requests with an unauthorized KMS key (e.g. using the Config rule: [S3\_BUCKET\_POLICY\_NOT\_MORE\_PERMISSIVE](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-policy-not-more-permissive.html), note that a new rule needs be deployed for each configuration, then the resource tracked by name or tag; alternatively you might use [S3\_BUCKET\_SERVER\_SIDE\_ENCRYPTION\_ENABLED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-server-side-encryption-enabled.html) to ensure a limited coverage) | Create a bucket not blocking PutObject requests with an unauthorized KMS key, it should be detected. | Medium | S3.FC1  S3.FC10  S3.FC5 | - | Very High |
| Detective (COSO)  Detect (NIST CSF) | [S3.C68, depends on S3.C61]  Monitor that only authorized KMS key(s) are used on each bucket (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Make a request encrypted with an unauthorized KMS key, it should be detected | Low | S3.FC5 | S3.T11 (Very Low)  S3.T16 (Low)  S3.T30 (Very Low)  S3.T36 (Low) | Low |
| Directive (COSO)  Identify (NIST CSF) | [S3.C145]  Maintain a list of buckets (or paths) required to be encrypted with server-side encryption with customer-provided keys (SSE-C) | Request the list of buckets (or paths) required to be encrypted with server-side encryption with customer-provided keys (SSE-C), its review process, and its review records | Medium | S3.FC10  S3.FC5 | S3.T11 (Very Low)  S3.T16 (Very Low)  S3.T20 (Very Low)  S3.T30 (Very Low)  S3.T36 (Very Low)  S3.T37 (Very Low) | Very High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C146, depends on S3.C145, assured by S3.C147]  For buckets (or paths) requiring SSE-C, block PutObject requests with unauthorized encryption (e.g. using an S3 bucket policy deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-customer-algorithm"="AES265" is not present) | Make a request to a bucket (or path) requiring SSE-C without the proper encryption, it should be denied | Low | S3.FC10  S3.FC5 | S3.T11 (High)  S3.T16 (High)  S3.T20 (Very High)  S3.T30 (High)  S3.T36 (High)  S3.T37 (High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C147]  For buckets (or paths) requiring SSE-C, verify all buckets block PutObject requests with unauthorized encryption | Create a bucket requiring SSE-C not blocking PutObject requests with unauthorized encryption, it should be detected. | High | S3.FC10  S3.FC5 | - | Very High |
| Detective (COSO)  Detect (NIST CSF) | [S3.C148, depends on S3.C145]  For buckets (or paths) requiring SSE-C, monitor that only authorized encryption is used on each bucket or path (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-side-encryption-customer-algorithm*) | Make a request to a bucket (or path) requiring SSE-C without the proper encryption, it should be detected | Low | S3.FC5 | S3.T11 (Very Low)  S3.T16 (Low)  S3.T30 (Very Low)  S3.T36 (Low) | Low |

## Protect primary data against loss [S3.CO17]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C69, depends on S3.C58, assured by S3.C70]  Enable versioning on buckets holding primary data | Request the mechanism used to ensure versioning on buckets holding primary data, and its records | Very Low | S3.FC1  S3.FC5 | S3.T16 (High)  S3.T17 (High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C70]  Verify buckets holding primary data are versioned (e.g. using [S3\_BUCKET\_VERSIONING\_ENABLED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-versioning-enabled.html)) | Remove versioning from a bucket holding primary data, it should be detected | Low | S3.FC1  S3.FC5 | - | High |
| Directive (COSO)  Recover (NIST CSF) | [S3.C71, depends on S3.C58]  Backup primary data in a secure location under a different security authority (e.g. in an [AWS data bunker account](https://wellarchitectedlabs.com/security/100_labs/100_create_a_data_bunker/1_instructions/) via replication, or using AWS Backup for Amazon S3) | Request the mechanism used to backup primary data in a location which have different security authority, its records of execution, and records of restoration testing | Medium | S3.FC1  S3.FC13  S3.FC5 | S3.T16 (High)  S3.T17 (High)  S3.T25 (High) | Medium |

## Encrypt or tokenize critical data [S3.CO18]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C72]  Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Request the governance and mechanism(s) used to protect data (e.g. encrypt or tokenize critical data on the client side) | Very High | S3.FC1  S3.FC10  S3.FC16  S3.FC5 | S3.T1 (Medium)  S3.T3 (Medium)  S3.T5 (High)  S3.T7 (High)  S3.T11 (Very High)  S3.T12 (Very High)  S3.T13 (Very High)  S3.T17 (High)  S3.T20 (High)  S3.T30 (High)  S3.T31 (High) | Medium |

## Have a process to apply legal hold [S3.CO19]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C73]  Create a process to apply a legal hold to any S3 bucket whenever required. The condition "s3:object-lock-legal-hold" can be used to restrict who can remove such a lock. | Request the process of applying legal hold, and its records | Medium | S3.FC1  S3.FC5 | S3.T16 (Low)  S3.T17 (Medium) | Medium |

## Use S3 Object Lock to protect data integrity [S3.CO20]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C74, depends on S3.C58, assured by S3.C75]  Implement the authorized default S3 Object Lock on each bucket (note: Amazon S3 evaluates and applies bucket policies before applying bucket default S3 Object Lock settings) | Upload an object without appropriate S3 Object Lock, it should have the S3 Object Lock automatically. | Low | S3.FC1  S3.FC13  S3.FC5 | S3.T16 (High)  S3.T17 (High)  S3.T25 (High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C75]  Verify all buckets have the correct default S3 Object Lock configuration | Create a bucket 1) without S3 Object Lock or 2) with an incorrect default S3 Object Lock, it should be detected. | Medium | S3.FC1  S3.FC13  S3.FC5 | - | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C76, depends on S3.C58, assured by S3.C77]  Block PutObject and PutObjectRetention requests with unauthorized S3 Object Lock on each bucket (e.g. using an S3 bucket policy deny statement on PutObject and PutObjectRetention if the condition if exists "s3:object-lock-mode" and "s3:object-lock-remaining-retention-days" is not the defined S3 Object Lock configuration) | Make a request with an incorrect S3 Object Lock configuration, it should be denied | Low | S3.FC1  S3.FC5 | S3.T16 (High)  S3.T17 (High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C77]  Verify all buckets blocks PutObject and PutObjectRetention requests with unauthorized S3 Object Lock (e.g. using the Config rule: [S3\_BUCKET\_POLICY\_NOT\_MORE\_PERMISSIVE](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-policy-not-more-permissive.html), note that a new rule needs be deployed for each configuration, then the resource tracked by name or tag) | Create a bucket not blocking PutObject and PutObjectRetention requests with unauthorized S3 Object Lock, it should be detected. | Medium | S3.FC1  S3.FC5 | - | High |

## Remove incomplete multipart uploads [S3.CO21]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C78, assured by S3.C79]  Reduce costs related to incomplete multipart upload by creating a lifecycle policy to remove them after an agreed length of time (e.g. 7 days) ([blog](https://aws.amazon.com/blogs/aws/s3-lifecycle-management-update-support-for-multipart-uploads-and-delete-markers/)) | Create an incomplete upload, and wait for the agreed time, it should be deleted automatically. | Low | S3.FC5 | S3.T40 (High) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C79]  Verify a lifecycle policy on incomplete multipart uploads is implemented on all buckets (e.g. using AWS Config rule: [S3\_LIFECYCLE\_POLICY\_CHECK](https://docs.aws.amazon.com/config/latest/developerguide/s3-lifecycle-policy-check.html)) | Create a bucket without a lifecycle policy to remove incomplete multipart upload, it should be detected | Medium | S3.FC5 | - | Low |

## Block deprecated actions [S3.CO22]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C80]  Block deprecated S3 actions using IAM ThreatModel and the S3 actions list. | Request the controls blocking deprecated S3 actions | Low | S3.FC1 | S3.T35 (Medium) | Very Low |

## Block all requests not using SigV4 [S3.CO23]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C81]  Block all requests not using SigV4 (e.g. using an SCP and S3 policy on all buckets with deny on "StringNotEquals":{"s3:signatureversion": "AWS4-HMAC-SHA256"}) | Make a non-SigV4 AWS API call, it should be denied. | Low | S3.FC1 | S3.T35 (High) | Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C82]  Monitor and investigate that all requests not using SigV4 (e.g. via CloudTrail S3 with the additionalEventData.SignatureVersion different from "SigV4") | Make a non-SigV4 AWS API call, it should be detected. | Low | S3.FC1 | S3.T35 (Low) | Very Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C83]  Use SDK with SigV4 enabled ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingAWSSDK.html#UsingAWSSDK-move-to-Sig4)) | Request the mechanism ensuring the use of SDK with SigV4 enabled | Low | S3.FC1 | S3.T35 (High) | Low |

## Block all requests not using HTTP authorization header, if not explicitly authorized [S3.CO24]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C84]  Block all requests not using HTTP authorization header, i.e. presign via query strings or POST ([ref](https://docs.aws.amazon.com/AmazonS3/latest/API/sig-v4-authenticating-requests.html)) (e.g. using an SCP and S3 policy on all buckets with deny on "StringNotEquals":{"s3:authType": "REST-HEADER"}). Note that it blocks uploads via the console, as well. | Make a request with a non-HTTP authorization header, it should be denied | Low | S3.FC5 | S3.T39 (Medium) | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C85]  Monitor and investigate that all requests not using SigV4 (e.g. via CloudTrail S3 with the additionalEventData.SignatureVersion different from "SigV4") | Make 1) a presigned AWS API call and 2) a POST request, it should be detected. | Low | S3.FC5 | S3.T39 (Very Low) | Very Low |

## Restrict bucket replication [S3.CO25]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C86]  Maintain a list of authorized buckets to have replication enabled, their target bucket and replication type (i.e. ownership, RTC, etc.) ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/replication-add-config.html)). | Request the list of authorized buckets to have replication enabled, their target bucket and replication rights, its review process, and its review records | Medium | S3.FC15 | S3.T2 (Very Low) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C134, depends on S3.C86, assured by S3.C87,S3.C88,S3.C117]  Ensure only authorized buckets have replication enabled and with correct configuration are configured | Request 1) the mechanism ensuring only authorized buckets have replication enabled and with correct configuration are configured, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Medium | S3.FC15 | S3.T2 (High) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C87]  Verify only authorized buckets have replication enabled and with correct configuration | Configure replication on a non-authorized bucket, it should be detected | Medium | S3.FC15 | - | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C88]  Verify authorized buckets have the correct replication configuration | Modify the configuration of an authorized replication, it should be detected | High | S3.FC15 | - | Medium |
| Directive (COSO)  Identify (NIST CSF) | [S3.C89]  Maintain a list of IAM roles used for replication, ideally dedicated (e.g. using change management process on infrastructure-as-code) | Request the list of all IAM roles configured for replication | Medium | S3.FC15 | S3.T2 (Very Low) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C138, depends on S3.C89, assured by S3.C92]  Ensure only authorized IAM roles are attached for each replication, ideally dedicated | Request the mechanism ensuring authorized IAM roles are attached for each replication, and the evidence of its execution for all replication configuration | Medium | S3.FC15 | S3.T2 (High) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C90, depends on S3.C89]  Limit the S3 access to the source/destination bucket and replication rights of each authorized IAM role configured for replication | Request the S3 access of replication role, and how they aligned to the replication requirements | Medium | S3.FC15 | S3.T2 (Medium) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C91, depends on S3.C89]  Limit access to authorized IAM roles used for replication, using the IAM ThreatModel (e.g. trust policy, and "iam:PassRole") | Request the IAM ThreatModel and the evidence of its application to the IAM roles used for replication | High | S3.FC15 | S3.T2 (High) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C92]  Verify only the authorized IAM role is configured for each replication | Create/modify a replication with an unauthorized IAM role, it should be detected | High | S3.FC15 | - | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C116]  Monitor abnormal behavior on replication CloudWatch metrics (i.e. *BytesPendingReplication* and *OperationsPendingReplication*) | Create an abnormal replication, it should be detected | Low | S3.FC15 | S3.T2 (Low) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C117]  Verify all replicated buckets have metrics enabled on each replication rule (included by default in S3 RTC) | Modify the replication metric of an authorized replication, it should be detected | Medium | S3.FC15 | - | Medium |

## Scan input/output objects for malware [S3.CO26]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Detective (COSO)  Detect (NIST CSF) | [S3.C93, depends on S3.C58]  If the bucket is used as an input or the output of a process, scan the objects for malware (e.g. using [VirusScan](https://s3-virusscan.widdix.net/), [Cloud Storage Security](https://cloudstoragesec.com/amazon-s3), or [Trend Micro Cloud One](https://aws.amazon.com/blogs/apn/amazon-s3-malware-scanning-using-trend-micro-cloud-one-and-aws-security-hub/)) | Inject a malware test file, it should be detected. | Medium | S3.FC16  S3.FC5 | S3.T14 (Medium)  S3.T15 (Low) | Medium |

## Control event receivers [S3.CO27]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C94]  Maintain a list of authorized notification receiver(s) (e.g. SNS topic, Lambda, etc.) for each bucket. You might use a simpler approach by using authorized account ID(s) to ensure all your receivers are in authorized AWS account(s). | Request the list of authorized notification receiver (e.g. SNS topic, Lambda, etc.) for each bucket, its review process, and its review records | Low | S3.FC20 | S3.T41 (Very Low) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C135, depends on S3.C94, assured by S3.C95]  Ensure only authorized notification receiver(s) (e.g. SNS topic, Lambda, etc.) for each bucket are configured | Request 1) the mechanism ensuring only authorized notification receiver(s) (e.g. SNS topic, Lambda, etc.) for each bucket are configured, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Medium | S3.FC20 | S3.T41 (High) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C95]  Verify only authorized notification receiver(s) are configured for buckets. | Create an unauthorized receiver, it should be detected. | High | S3.FC20 | - | Low |

## Control where the inventory is stored [S3.CO28]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C96]  Maintain a list of authorized S3 buckets to receive S3 inventory of each bucket | Request the list of authorized bucket(s) to receive S3 inventory of each bucket, its review process, and its review records | Low | S3.FC12 | S3.T42 (Very Low) | Very Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C136, depends on S3.C96, assured by S3.C97]  Ensure only authorized S3 buckets are configured to receive S3 inventory for each bucket | Request 1) the mechanism ensuring only authorized S3 buckets are configured to receive S3 inventory for each bucket, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Medium | S3.FC12 | S3.T42 (Medium) | Very Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C97]  Verify only authorized buckets are configured to receive inventory. | Create an unauthorized bucket to receive inventory, it should be detected. | High | S3.FC12 | - | Very Low |

## Limit access from only authorized VPCs [S3.CO29]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C98]  For each S3 bucket, maintain a list of VPC(s) authorized to access it. | For each S3 bucket, request the list of authorized VPC to access it, its review process, and its review records | Low | S3.FC1  S3.FC10  S3.FC2  S3.FC5 | S3.T11 (Very Low)  S3.T14 (Very Low)  S3.T17 (Very Low)  S3.T30 (Very Low)  S3.T33 (Very Low)  S3.T36 (Very Low)  S3.T38 (Very Low)  S3.T39 (Very Low) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C99, depends on S3.C98, assured by S3.C100]  Limit the access to only those VPC(s) (e.g. using S3 bucket statement, deny if the condition "aws:SourceVpce", or if the bucket policy size is beyond the limit, use this condition on access point) | Make a request to the bucket outside an authorized VPC, it should be denied | Very Low | S3.FC1  S3.FC10  S3.FC2  S3.FC5 | S3.T11 (Medium)  S3.T14 (Medium)  S3.T17 (Medium)  S3.T30 (High)  S3.T33 (High)  S3.T36 (High)  S3.T38 (High)  S3.T39 (High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C100]  Verify all buckets include a control to limit access to only authorized VPC(s) (e.g. using the AWS Config rule [S3\_BUCKET\_POLICY\_GRANTEE\_CHECK](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-policy-grantee-check.html)) | Remove the control limiting access to only authorized VPC(s), it should be detected. | Medium | S3.FC1  S3.FC10  S3.FC2  S3.FC5 | - | High |

## Control CloudFront access [S3.CO30]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C101]  Maintain a list of authorized CloudFront distribution (via Origin Access Control) and associated bucket | Request the list of all authorized CloudFront distribution and associated S3 buckets | Low | S3.FC10 | S3.T20 (Very Low) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C137, depends on S3.C101, assured by S3.C102]  Ensure only authorized CloudFront distributions are associated with their authorized bucket, and vice versa. | Request 1) the mechanism ensuring only authorized only authorized CloudFront distribution are associated with their authorized bucket, and vice versa, 2) its records of execution for all new CloudFront distribution, and 3) plan to move any older CloudFront distribution | Medium | S3.FC10 | S3.T20 (High) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C102]  Verify all associations of CloudFront distributions with buckets are authorized | Create a non-authorized distribution or association, it should be detected. | High | S3.FC10 | - | Medium |

## Protect and/or claim your domains and trademarks/copyrights [S3.CO31]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C103]  Protect and/or claim your domains and trademarks/copyrights (by creating your trademark buckets and using the [copyright infringement process](https://aws.amazon.com/terms/#notice-and-procedure-for-making-claims-of-copyright-infringement) from AWS) | Request the process by protecting and/or claiming your domains and trademarks/copyrights | Medium | S3.FC28 | S3.T23 (High) | Low |

## Restrict access point access to VPC when in use [S3.CO32]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C104]  Maintain a list of authorized access between VPCs, S3 access points, and S3. | Request the list of authorized access between VPC and S3 access points | Medium | S3.FC1  S3.FC26  S3.FC28  S3.FC5 | S3.T7 (Very Low)  S3.T9 (Very Low)  S3.T10 (Very Low)  S3.T11 (Very Low)  S3.T28 (Very Low) | Medium |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C105, depends on S3.C104, assured by S3.C109]  Limit access via the S3 access point by using a VPC endpoint and/or bucket policy with the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" in an allow statement to reduce the length of the allowlist bucket name in the VPC endpoint/bucket policy. | Do a request on an unauthorized access point or bucket, it should be denied. | Medium | S3.FC1  S3.FC26  S3.FC28  S3.FC33  S3.FC5 | S3.T7 (Medium)  S3.T9 (Very High)  S3.T10 (Very High)  S3.T11 (Medium)  S3.T54 (Medium)  S3.T55 (Medium) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C106, assured by S3.C110]  In the S3 bucket policy, deny all IAM principals not using an authorized S3 access point(s) using the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" | Query the bucket outside S3 access point, it should be denied. | Medium | S3.FC1  S3.FC26  S3.FC33 | S3.T7 (Medium)  S3.T28 (High)  S3.T55 (Medium)  S3.T56 (Very High) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C107]  Block the creation "s3:CreateAccessPoint" of non-VPC S3 access point (e.g. using the condition "StringNotEquals": {"s3:AccessPointNetworkOrigin": "VPC"}) | Do a request to create an internet-based access point, it should be denied. | Low | S3.FC1  S3.FC26 | S3.T7 (Medium)  S3.T28 (Very High) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C108, assured by S3.C111]  Block all traffic from Internet-configured S3 access point (e.g. on the bucket policy, using a deny statement with the condition "StringNotEquals": {"s3:AccessPointNetworkOrigin": "VPC"}) | Create an internet-facing access point and try to access a bucket, it should be denied. | Low | S3.FC1  S3.FC26  S3.FC28 | S3.T7 (Medium)  S3.T10 (Medium)  S3.T28 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C109]  Verify only access points are used in the resource-level statement in VPC endpoints | Create a VPC endpoint giving access to an S3 bucket, it should be detected. | High | S3.FC1  S3.FC26  S3.FC28  S3.FC33  S3.FC5 | - | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C110]  Verify S3 bucket policies deny non-authorized S3 access points | Remove/modify the deny on the bucket policy, it should be detected. | High | S3.FC1  S3.FC26  S3.FC33 | - | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C111]  Verify all S3 access points are VPC attached | Create an internet-based access point, it should be detected. | Low | S3.FC1  S3.FC26  S3.FC28 | - | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C112, depends on S3.C104]  Block any [object-related operations](https://docs.aws.amazon.com/AmazonS3/latest/dev/using-access-points.html#access-points-service-api-support) access to S3 bucket not through access point (i.e. using a deny IAM policy statement with the condition "ArnNotLike" {"s3:DataAccessPointArn": "arn:aws:s3:*Region*:*AccountId*:accesspoint/\*"}) | Access any S3 bucket using the S3 public endpoint, it should be denied. | Low | S3.FC1  S3.FC5 | S3.T7 (Medium)  S3.T11 (High) | High |

## Control IAM roles used for Batch [S3.CO33]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C120]  Maintain a list of IAM roles used for Batch job, ideally dedicated (e.g. using change management process on infrastructure-as-code) | Request the list of all IAM roles configured for Batch job | Medium | S3.FC27 | S3.T44 (Very Low) | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C139, depends on S3.C120, assured by S3.C123]  Ensure only an authorized IAM role is attached on each Batch job | Request the mechanism ensuring only an authorized IAM role is attached on each Batch job, and the evidence of its execution for all new {resource} | Medium | S3.FC27 | S3.T44 (High) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C121, depends on S3.C120]  Limit the access to only required resources/permissions (e.g. source/destination bucket, Lambda functions) of each authorized IAM role configured for Batch jobs | Request the access to only required resources/permissions for each Batch IAM role, and how they aligned to the replication requirements | Medium | S3.FC27 | S3.T44 (High) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C122, depends on S3.C120]  Limit access to authorized IAM roles used for Batch job, using the IAM ThreatModel (e.g. trust policy, and "iam:PassRole") | Request the IAM ThreatModel and the evidence of its application to the IAM roles used for Batch job | Medium | S3.FC27 | S3.T44 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C123]  Verify only the authorized IAM role is configured for each Batch job | Create/modify a Batch job with an unauthorized IAM role, it should be detected | High | S3.FC27 | - | Medium |

## Enforce only authorized Object Lambda access point and associated access [S3.CO34]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C125]  Maintain a list of authorized Lambda function for each Object Lambda access point, its associated access point, its associated GET request(s), and payload | Request the list of authorized Lambda function for each Object Lambda access point, its associated access point, its associated GET request(s), and payload, its review process, and its review records | Low | S3.FC32 | S3.T46 (Very Low) | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C126, depends on S3.C125, assured by S3.C127]  Ensure only authorized Lambda function for each Object Lambda access point, its associated access point, its associated GET request(s), and payload are created | Request the mechanism ensuring only authorized Lambda function for each Object Lambda access point, its associated access point, its associated GET request(s), and payload, and the evidence of its execution | Medium | S3.FC32 | S3.T46 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C127]  Verify only the authorized Lambda function are configured on each Object Lambda access point, its associated access point, its associated GET request(s), and payload | Attach 1) an unauthorized Lambda function on an Object Lambda access point, 2) an unauthorized Object Lambda access point to an access point, 3) an authorized Lambda function with an unauthorized GET request on an Object Lambda access point, and 4) an authorized Lambda function with an unauthorized payload, it should be detected. | Medium | S3.FC32 | - | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C128]  Ensure Lambda functions configured on Object Lambda access point are secured using Lambda ThreatModel | Request the mechanism ensuring Lambda ThreatModel and its application for Lambda functions associated to Object Lambda access point, and its records of execution | Medium | S3.FC32 | S3.T46 (High) | Medium |
| Directive (COSO)  Identify (NIST CSF) | [S3.C129]  Maintain a list of cross-account access on each Object Lambda access point | Request the list of authorized cross-account access for each Object Lambda access point, its review process, and its review records | Very Low | S3.FC32 | S3.T46 (Very Low) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C130, depends on S3.C129, assured by S3.C131]  Ensure only authorized cross-account IAM entities are allowed in the Object Lambda access point policy | Request the mechanism ensuring only cross-account IAM entities are allowed in the Object Lambda access point policy, and the evidence of its execution | Low | S3.FC32 | S3.T46 (Medium) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C131]  Verify only the authorized cross-account IAM entities are allowed in the Object Lambda access point policy | Add 1) an unauthorized cross-account IAM entity on an Object Lambda access point policy, it should be detected. | High | S3.FC32 | - | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C132, assured by S3.C133]  Ensure CloudWatch is enabled for all Object Lambda access points | Request the mechanism ensuring CloudWatch is enabled for all Object Lambda access points, and its records of execution | Low | S3.FC32 | S3.T46 (Low) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C133]  Verify CloudWatch is enabled for all Object Lambda access points | Create an Object Lambda access point without CloudWatch enabled, it should be detected. | Low | S3.FC32 | - | Low |

## Deploy only authorized S3 website and are placed behind a CDN [S3.CO35]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Identify (NIST CSF) | [S3.C141]  Maintain a list of authorized buckets to be configured as a S3 website endpoint | Request the list of authorized buckets to be configured as a website endpoint, its review process, and its review records | Low | S3.FC16 | S3.T13 (Very Low)  S3.T29 (Very Low) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C142, depends on S3.C141, assured by S3.C143]  Ensure only authorized buckets are configured as a S3 website endpoint | Request 1) the mechanism ensuring only authorized buckets are configured as a S3 website endpoint, 2) its records of execution for all new website-enabled buckets, and 3) plan to move any older website-enabled buckets | Medium | S3.FC16 | S3.T13 (Medium)  S3.T29 (Medium) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C143]  Verify only authorized buckets are configured as S3 website endpoints | Enable static website hosting on an unauthorized bucket, it should be detected. | Medium | S3.FC16 | - | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C144, depends on S3.C141]  Ensure S3 website endpoints are protected with HTTP headers ([ref](https://aws.amazon.com/blogs/networking-and-content-delivery/adding-http-security-headers-using-lambdaedge-and-amazon-cloudfront/)) using a CDN (e.g. CloudFront) | Request the mechanism ensuring S3 website endpoints are protected with HTTP headers | Medium | S3.FC16 | S3.T13 (High)  S3.T29 (Very High) | High |

## Use an unguessable naming convention [S3.CO36]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C29]  Use an unguessable naming convention for the email addresses of your AWS accounts (e.g. add a + sign and a random string to redirect the email in the same mailbox) | Review naming convention for root account email and their implementation | Medium | S3.FC28 | S3.T19 (High) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C30]  Use an unguessable naming convention for your IAM users and IAM roles (e.g. add a random string) | Review naming convention for IAM users/role and their implementation | Medium | S3.FC28 | S3.T24 (High) | Low |

## Disabling ACLs for all buckets [S3.CO37]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C152, assured by S3.C154]  Ensure bucket ACL and object ACL are disabled on each bucket | Request 1) the mechanism ensuring bucket ACL and object ACL are disabled on each bucket, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Medium | S3.FC1  S3.FC24  S3.FC25  S3.FC5  S3.FC8 | S3.T4 (Very High)  S3.T6 (Very High)  S3.T36 (Very High)  S3.T43 (Very High)  S3.T52 (Very High)  S3.T53 (Very High) | Very High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C153]  Prevent the creation of buckets with ACL enabled (e.g. by using a SCP and/or an IAM policy on "s3:CreateBucket" with a deny statement on StringNotEquals "s3:x-amz-object-ownership":"BucketOwnerEnforced"). Note that it does not block someone from enabling an ACL afterward via PutPutBucketOwnershipControls. | Create a bucket to enable ACL, it should be denied. | Low | S3.FC1  S3.FC24  S3.FC25  S3.FC5  S3.FC8 | S3.T4 (High)  S3.T6 (High)  S3.T36 (High)  S3.T43 (High)  S3.T52 (High)  S3.T53 (High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C154]  Verify bucket ACL and object ACL are disabled on each bucket (e.g. using the AWS Config rule [S3\_BUCKET\_ACL\_PROHIBITED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-acl-prohibited.html) for bucket ACL) | Create/modify a bucket to enable ACL, it should be detected. | Medium | S3.FC1  S3.FC24  S3.FC25  S3.FC5  S3.FC8 | - | Very High |

## Ensure all requests are blocked from unauthorized service roles [S3.CO38]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C155, assured by S3.C156]  Ensure all requests are blocked from unauthorized service roles (e.g. by denying all requests with the principal "arn:aws:iam::\*:\*/AWSServiceRoleFor\*" on S3 bucket policies) | Request 1) the mechanism ensuring only authorized service roles can access each bucket, 2) its records of execution for all new bucket, and 3) plan to move any older bucket | Low | S3.FC28 | S3.T57 (Very High) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C156]  Verify all requests are blocked from unauthorized service roles. | Remove the statement on a S3 bucket policy denying all unauthorized service roles, it should be detected. | Medium | S3.FC28 | - | Low |

## Enforce S3 access logging [S3.CO39]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Detective (COSO)  Detect (NIST CSF) | [S3.C157]  Monitor PutBucketLogging to detect bucket logging changes including deactivation and bucket change (i.e. using CloudTrail event *PutBucketLogging* and *requestParameter.BucketLoggingStatus* field to examine the lack of "LoggingEnabled" key, or an unauthorized bucket in requestParameters.BucketLoggingStatus.LoggingEnabled.TargetBucket) | Make a call to 1) disable bucket logging, or 2) change to an unauthorized bucket, it should be detected. | Low | S3.FC19 | S3.T59 (Medium) | Very Low |

# Compliance Mapping

## PCI DSS v3.2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **PCI DSS v3.2** | **Control Objectives** | **Controls** | | | | |
| **Very High** | **High** | **Medium** | **Low** | **Very Low** |
| 1.1  1.1.1  1.1.2  1.1.3  1.1.4  1.1.5  1.1.6  1.1.7  1.2  1.2.1  1.2.2  1.2.3  1.3 | [S3.CO13] Block direct public access  [S3.CO30] Control CloudFront access | S3.C49  S3.C50  S3.C51  S3.C52  S3.C53  S3.C54 | - | S3.C47  S3.C48  S3.C101  S3.C102  S3.C137 | - | - |
| 1.3.1  1.3.2 | [S3.CO13] Block direct public access | S3.C49  S3.C50  S3.C51  S3.C52  S3.C53  S3.C54 | - | S3.C47  S3.C48 | - | - |
| 1.3.3 | [S3.CO13] Block direct public access  [S3.CO30] Control CloudFront access | S3.C49  S3.C50  S3.C51  S3.C52  S3.C53  S3.C54 | - | S3.C47  S3.C48  S3.C101  S3.C102  S3.C137 | - | - |
| 1.3.4 | [S3.CO13] Block direct public access | S3.C49  S3.C50  S3.C51  S3.C52  S3.C53  S3.C54 | - | S3.C47  S3.C48 | - | - |
| 1.3.5  1.3.6  1.3.7 | [S3.CO5] Identify and ensure the protection of all external buckets hosting your objects  [S3.CO15] Identify and ensure the protection of all internal buckets hosting your objects | S3.C11 | S3.C15  S3.C114  S3.C58 | S3.C12  S3.C13  S3.C14  S3.C115  S3.C59 | - | S3.C60 |
| 2.1  2.1.1 | [S3.CO16] Enforce encryption-at-rest  [S3.CO18] Encrypt or tokenize critical data | S3.C61  S3.C64  S3.C65  S3.C66  S3.C67  S3.C145  S3.C146  S3.C147 | - | S3.C62  S3.C140  S3.C72 | S3.C63  S3.C68  S3.C148 | - |
| 2.2.3 | [S3.CO9] Block requests with KMS keys from unauthorized AWS account(s)  [S3.CO16] Enforce encryption-at-rest  [S3.CO18] Encrypt or tokenize critical data | S3.C61  S3.C64  S3.C65  S3.C66  S3.C67  S3.C145  S3.C146  S3.C147 | S3.C31  S3.C32 | S3.C62  S3.C140  S3.C72 | S3.C33  S3.C63  S3.C68  S3.C148 | - |
| 3.5  3.5.1  3.5.2  3.5.3  3.5.4  3.6  3.6.1  3.6.2  3.6.3  3.6.4  3.6.5  3.6.6  3.6.7  3.6.8 | [S3.CO16] Enforce encryption-at-rest  [S3.CO18] Encrypt or tokenize critical data | S3.C61  S3.C64  S3.C65  S3.C66  S3.C67  S3.C145  S3.C146  S3.C147 | - | S3.C62  S3.C140  S3.C72 | S3.C63  S3.C68  S3.C148 | - |
| 4.1 | [S3.CO26] Scan input/output objects for malware | - | - | S3.C93 | - | - |
| 5.1  5.1.1  5.1.2  5.2  5.3  5.4 | [S3.CO27] Control event receivers  [S3.CO28] Control where the inventory is stored | - | - | - | S3.C94  S3.C95  S3.C135 | S3.C96  S3.C97  S3.C136 |
| 6.4  6.4.1  6.4.2  6.4.3  6.4.4  6.4.5  6.4.5.1  6.4.5.2  6.4.5.3  6.4.5.4  6.4.6 | [S3.CO12] Enforce good coding practice | - | S3.C113 | S3.C44  S3.C46 | S3.C41  S3.C42 | S3.C43  S3.C45 |
| 6.6 | [S3.CO8] Limit the access to the IAM actions required to execute the threats  [S3.CO10] Block changes to make an object public via object ACL  [S3.CO13] Block direct public access  [S3.CO33] Control IAM roles used for Batch  [S3.CO34] Enforce only authorized Object Lambda access point and associated access  [S3.CO35] Deploy only authorized S3 website and are placed behind a CDN  [S3.CO37] Disabling ACLs for all buckets  [S3.CO38] Ensure all requests are blocked from unauthorized service roles | S3.C149  S3.C150  S3.C151  S3.C49  S3.C50  S3.C51  S3.C52  S3.C53  S3.C54  S3.C152  S3.C153  S3.C154 | S3.C26  S3.C34  S3.C36  S3.C120  S3.C122  S3.C125  S3.C126  S3.C127  S3.C144 | S3.C27  S3.C28  S3.C47  S3.C48  S3.C121  S3.C123  S3.C139  S3.C128  S3.C129  S3.C130  S3.C131  S3.C141  S3.C142  S3.C143 | S3.C35  S3.C37  S3.C132  S3.C133  S3.C155  S3.C156 | - |
| 7.1  7.1.1  7.1.2  7.1.3  7.1.4  7.2  7.2.1 | [S3.CO13] Block direct public access | S3.C49  S3.C50  S3.C51  S3.C52  S3.C53  S3.C54 | - | S3.C47  S3.C48 | - | - |
| 7.2.2 | [S3.CO8] Limit the access to the IAM actions required to execute the threats  [S3.CO10] Block changes to make an object public via object ACL  [S3.CO13] Block direct public access  [S3.CO33] Control IAM roles used for Batch  [S3.CO34] Enforce only authorized Object Lambda access point and associated access  [S3.CO35] Deploy only authorized S3 website and are placed behind a CDN  [S3.CO37] Disabling ACLs for all buckets  [S3.CO38] Ensure all requests are blocked from unauthorized service roles | S3.C149  S3.C150  S3.C151  S3.C49  S3.C50  S3.C51  S3.C52  S3.C53  S3.C54  S3.C152  S3.C153  S3.C154 | S3.C26  S3.C34  S3.C36  S3.C120  S3.C122  S3.C125  S3.C126  S3.C127  S3.C144 | S3.C27  S3.C28  S3.C47  S3.C48  S3.C121  S3.C123  S3.C139  S3.C128  S3.C129  S3.C130  S3.C131  S3.C141  S3.C142  S3.C143 | S3.C35  S3.C37  S3.C132  S3.C133  S3.C155  S3.C156 | - |
| 7.2.3 | [S3.CO5] Identify and ensure the protection of all external buckets hosting your objects  [S3.CO15] Identify and ensure the protection of all internal buckets hosting your objects | S3.C11 | S3.C15  S3.C114  S3.C58 | S3.C12  S3.C13  S3.C14  S3.C115  S3.C59 | - | S3.C60 |
| 8.1.2 | [S3.CO1] Enforce encryption-in-transit | - | S3.C1  S3.C2  S3.C6  S3.C7  S3.C119 | S3.C3  S3.C5 | S3.C4 | - |
| 8.1.5 | [S3.CO5] Identify and ensure the protection of all external buckets hosting your objects  [S3.CO15] Identify and ensure the protection of all internal buckets hosting your objects | S3.C11 | S3.C15  S3.C114  S3.C58 | S3.C12  S3.C13  S3.C14  S3.C115  S3.C59 | - | S3.C60 |
| 8.2  8.2.1  8.2.2  8.2.3  8.2.4  8.2.5  8.2.6  8.3 | [S3.CO8] Limit the access to the IAM actions required to execute the threats  [S3.CO10] Block changes to make an object public via object ACL  [S3.CO13] Block direct public access  [S3.CO33] Control IAM roles used for Batch  [S3.CO34] Enforce only authorized Object Lambda access point and associated access  [S3.CO35] Deploy only authorized S3 website and are placed behind a CDN  [S3.CO37] Disabling ACLs for all buckets  [S3.CO38] Ensure all requests are blocked from unauthorized service roles | S3.C149  S3.C150  S3.C151  S3.C49  S3.C50  S3.C51  S3.C52  S3.C53  S3.C54  S3.C152  S3.C153  S3.C154 | S3.C26  S3.C34  S3.C36  S3.C120  S3.C122  S3.C125  S3.C126  S3.C127  S3.C144 | S3.C27  S3.C28  S3.C47  S3.C48  S3.C121  S3.C123  S3.C139  S3.C128  S3.C129  S3.C130  S3.C131  S3.C141  S3.C142  S3.C143 | S3.C35  S3.C37  S3.C132  S3.C133  S3.C155  S3.C156 | - |
| 8.7 | [S3.CO7] Limit and monitor access via S3 VPC endpoints  [S3.CO8] Limit the access to the IAM actions required to execute the threats  [S3.CO11] Prevent deletion of buckets  [S3.CO14] Block bucket ACL  [S3.CO22] Block deprecated actions  [S3.CO23] Block all requests not using SigV4  [S3.CO24] Block all requests not using HTTP authorization header, if not explicitly authorized  [S3.CO25] Restrict bucket replication  [S3.CO29] Limit access from only authorized VPCs  [S3.CO32] Restrict access point access to VPC when in use  [S3.CO33] Control IAM roles used for Batch  [S3.CO34] Enforce only authorized Object Lambda access point and associated access  [S3.CO35] Deploy only authorized S3 website and are placed behind a CDN  [S3.CO37] Disabling ACLs for all buckets  [S3.CO38] Ensure all requests are blocked from unauthorized service roles | S3.C149  S3.C150  S3.C151  S3.C152  S3.C153  S3.C154 | S3.C17  S3.C19  S3.C20  S3.C26  S3.C38  S3.C39  S3.C55  S3.C57  S3.C98  S3.C99  S3.C100  S3.C105  S3.C106  S3.C107  S3.C108  S3.C109  S3.C110  S3.C111  S3.C112  S3.C120  S3.C122  S3.C125  S3.C126  S3.C127  S3.C144 | S3.C22  S3.C23  S3.C24  S3.C27  S3.C28  S3.C84  S3.C86  S3.C87  S3.C88  S3.C89  S3.C90  S3.C91  S3.C92  S3.C117  S3.C134  S3.C138  S3.C104  S3.C121  S3.C123  S3.C139  S3.C128  S3.C129  S3.C130  S3.C131  S3.C141  S3.C142  S3.C143 | S3.C18  S3.C25  S3.C124  S3.C56  S3.C81  S3.C83  S3.C116  S3.C132  S3.C133  S3.C155  S3.C156 | S3.C21  S3.C40  S3.C80  S3.C82  S3.C85 |
| 8.18.4 | [S3.CO3] Enable CloudTrail S3 data events  [S3.CO4] Monitor S3 with Amazon GuardDuty and Macie  [S3.CO17] Protect primary data against loss  [S3.CO25] Restrict bucket replication  [S3.CO39] Enforce S3 access logging | - | S3.C118  S3.C69  S3.C70 | S3.C9  S3.C10  S3.C71  S3.C86  S3.C87  S3.C88  S3.C89  S3.C90  S3.C91  S3.C92  S3.C117  S3.C134  S3.C138 | S3.C116 | S3.C157 |
| 10.1 | [S3.CO3] Enable CloudTrail S3 data events  [S3.CO4] Monitor S3 with Amazon GuardDuty and Macie  [S3.CO8] Limit the access to the IAM actions required to execute the threats  [S3.CO17] Protect primary data against loss  [S3.CO25] Restrict bucket replication  [S3.CO29] Limit access from only authorized VPCs  [S3.CO33] Control IAM roles used for Batch  [S3.CO34] Enforce only authorized Object Lambda access point and associated access  [S3.CO35] Deploy only authorized S3 website and are placed behind a CDN  [S3.CO38] Ensure all requests are blocked from unauthorized service roles | S3.C149  S3.C150  S3.C151 | S3.C118  S3.C26  S3.C69  S3.C70  S3.C98  S3.C99  S3.C100  S3.C120  S3.C122  S3.C125  S3.C126  S3.C127  S3.C144 | S3.C9  S3.C10  S3.C27  S3.C28  S3.C71  S3.C86  S3.C87  S3.C88  S3.C89  S3.C90  S3.C91  S3.C92  S3.C117  S3.C134  S3.C138  S3.C121  S3.C123  S3.C139  S3.C128  S3.C129  S3.C130  S3.C131  S3.C141  S3.C142  S3.C143 | S3.C116  S3.C132  S3.C133  S3.C155  S3.C156 | - |
| 10.2  10.2.1  10.2.2  10.2.3  10.2.4  10.2.5  10.2.6  10.2.7 | [S3.CO3] Enable CloudTrail S3 data events  [S3.CO4] Monitor S3 with Amazon GuardDuty and Macie  [S3.CO17] Protect primary data against loss  [S3.CO25] Restrict bucket replication | - | S3.C118  S3.C69  S3.C70 | S3.C9  S3.C10  S3.C71  S3.C86  S3.C87  S3.C88  S3.C89  S3.C90  S3.C91  S3.C92  S3.C117  S3.C134  S3.C138 | S3.C116 | - |
| 10.3  10.3.1  10.3.2  10.3.3  10.3.4  10.3.5  10.3.6 | [S3.CO3] Enable CloudTrail S3 data events  [S3.CO4] Monitor S3 with Amazon GuardDuty and Macie  [S3.CO8] Limit the access to the IAM actions required to execute the threats  [S3.CO11] Prevent deletion of buckets  [S3.CO17] Protect primary data against loss  [S3.CO25] Restrict bucket replication  [S3.CO33] Control IAM roles used for Batch  [S3.CO38] Ensure all requests are blocked from unauthorized service roles  [S3.CO39] Enforce S3 access logging | S3.C149  S3.C150  S3.C151 | S3.C118  S3.C26  S3.C38  S3.C39  S3.C69  S3.C70  S3.C120  S3.C122 | S3.C9  S3.C10  S3.C27  S3.C28  S3.C71  S3.C86  S3.C87  S3.C88  S3.C89  S3.C90  S3.C91  S3.C92  S3.C117  S3.C134  S3.C138  S3.C121  S3.C123  S3.C139 | S3.C116  S3.C155  S3.C156 | S3.C40  S3.C157 |
| 10.5  10.5.1  10.5.2  10.5.3  10.5.4  10.5.5 | [S3.CO3] Enable CloudTrail S3 data events  [S3.CO4] Monitor S3 with Amazon GuardDuty and Macie  [S3.CO25] Restrict bucket replication  [S3.CO39] Enforce S3 access logging | - | S3.C118 | S3.C9  S3.C10  S3.C86  S3.C87  S3.C88  S3.C89  S3.C90  S3.C91  S3.C92  S3.C117  S3.C134  S3.C138 | S3.C116 | S3.C157 |
| 10.6  10.6.1  10.6.2  10.6.3  10.8  10.8.1  11.1  11.5  11.5.1 | [S3.CO1] Enforce encryption-in-transit | - | S3.C1  S3.C2  S3.C6  S3.C7  S3.C119 | S3.C3  S3.C5 | S3.C4 | - |
| 12.3.8  12.3.9 | [S3.CO6] Model the threats on all AWS services accessing S3 | - | - | S3.C16 | - | - |

*The Control Objectives are mapped to the* [*Secure Controls Framework*](https://www.securecontrolsframework.com/secure-controls-framework) *(SCF), provided under Attribution-NoDerivatives 4.0 International (CC BY-ND 4.0). Compliance mappings are not designed to fully ensure compliance with a specific governance or compliance standard. You are responsible for making your own assessment of whether your use of the Services meets applicable legal and regulatory requirements.*

*You can change the displayed Compliance mappings by contacting* [*chatbot@trustoncloud.com*](mailto:chatbot@trustoncloud.com)*.*

# Appendixes

## Appendix 1 - Prioritized list for control implementation

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Type** | **Control** | **Testing** | **Effort** | **Feature**  **Class(es)** | **Threat(s)**  **and Impact** | **CVSS-weighted**  **Priority** |
| Directive (COSO)  Protect (NIST CSF) | [S3.C11]  Track all buckets you don't control hosting your objects, define their authorized data classification, identify their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), and get assured of the protection (e.g. through contractual agreement, verified by assurance programs, or using this ThreatModel). | Request the list of all authorized external buckets authorized to host your objects, their respective owners (and AWS account ID), their ObjectACL requirements (including S3 Object Ownership), their data classification and the mechanism used to ensure the security of those buckets | Medium | S3.FC1  S3.FC16  S3.FC5 | S3.T3 (High)  S3.T5 (Very Low)  S3.T6 (Low)  S3.T7 (Very Low)  S3.T8 (Very Low)  S3.T9 (Very Low)  S3.T11 (Low)  S3.T14 (Very Low)  S3.T15 (Very Low)  S3.T21 (Very Low)  S3.T31 (High)  S3.T43 (Very High) | Very High |
| Directive (COSO)  Identify (NIST CSF) | [S3.C149]  For each bucket, maintain a list of authorized IAM principals allowed to access via bucket policy | Request the list of authorized a list of authorized IAM principals allowed to access via bucket policy, its review process, and its review records | Medium | S3.FC10 | S3.T37 (Very Low) | Very High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C150, depends on S3.C149, assured by S3.C151]  Ensure only authorized a list of authorized IAM principals allowed to access via bucket policy are configured (e.g. using IAM Access Analyzer for the reconciliation) | Request 1) the mechanism ensuring only authorized IAM principals allowed to access via bucket policy are configured, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Medium | S3.FC10 | S3.T37 (Very High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C151]  Verify only authorized IAM principals allowed to access via bucket policy are used (e.g. using the AWS Config rule [S3\_BUCKET\_POLICY\_GRANTEE\_CHECK](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-policy-grantee-check.html)) | Allow an unauthorized IAM principal on a bucket policy, it should be detected. | Medium | S3.FC10 | - | Very High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C49, assured by S3.C50]  Enable account-level S3 Block Public Access on all AWS accounts, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Request 1) the mechanism ensuring account-level S3 Block Public Access is enabled on all AWS accounts, 2) its records of execution for all new AWS accounts, and 3) plan to move any older AWS accounts | Very Low | S3.FC10  S3.FC5  S3.FC8 | S3.T4 (High)  S3.T14 (High)  S3.T36 (Very High)  S3.T37 (Very High)  S3.T38 (Medium) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C50]  Verify account-level S3 Block Public Access is enabled on all AWS accounts, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true (e.g. using the AWS Config rule: [S3\_ACCOUNT\_LEVEL\_PUBLIC\_ACCESS\_BLOCKS](https://docs.aws.amazon.com/config/latest/developerguide/s3-account-level-public-access-blocks.html)). | Remove the account-level S3 Block Public Access of an AWS account, it should be detected. | Very Low | S3.FC10  S3.FC5  S3.FC8 | - | Very High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C51, assured by S3.C52]  Enable S3 Block Public Access on all S3 buckets, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Request 1) the mechanism ensuring S3 Block Public Access is enabled on each bucket, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Low | S3.FC10  S3.FC5  S3.FC8 | S3.T4 (High)  S3.T14 (High)  S3.T36 (Very High)  S3.T37 (Very High)  S3.T38 (Medium) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C52]  Verify S3 Block Public Access is enabled on all S3 buckets, with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true (e.g. using the AWS Config rule: [S3\_BUCKET\_LEVEL\_PUBLIC\_ACCESS\_PROHIBITED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-level-public-access-prohibited.html)). | Remove a S3 Block Public Access of an S3 bucket, it should be detected. | Very Low | S3.FC10  S3.FC5  S3.FC8 | - | Very High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C53, assured by S3.C54]  Enable S3 Block Public Access on all S3 access points (including multi-region), with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Request 1) the mechanism ensuring S3 Block Public Access is enabled on each S3 access point, 2) its records of execution for all new S3 access points, and 3) plan to move any older S3 access points | Low | S3.FC10  S3.FC26  S3.FC33  S3.FC5 | S3.T14 (High)  S3.T36 (Medium)  S3.T37 (Medium)  S3.T38 (Medium)  S3.T54 (High)  S3.T55 (High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C54]  Verify S3 Block Public Access is enabled on all S3 access points (including multi-region), with BlockPublicAcls, IgnorePublicAcls, BlockPublicPolicy, and RestrictPublicBuckets set to true. | Remove S3 Block Public Access of 1) an access point, and 2) a Multi-Region Access Point, it should be detected | Medium | S3.FC10  S3.FC26  S3.FC33  S3.FC5 | - | Very High |
| Directive (COSO)  Identify (NIST CSF) | [S3.C61]  Maintain a list of authorized KMS key(s) for each bucket and their default encryption key. You might simplify by using only 1 key per bucket, ideally dedicated. Note that an S3 server access log bucket does not support KMS encryption ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/bucket-encryption.html#bucket-encryption-how-to-set-up)). | Request the list of authorized KMS key(s) for each bucket, its review process, and its review records | Medium | S3.FC1  S3.FC10  S3.FC5 | S3.T11 (Very Low)  S3.T16 (Very Low)  S3.T17 (Very Low)  S3.T20 (Very Low)  S3.T30 (Very Low)  S3.T36 (Very Low)  S3.T37 (Very Low) | Very High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C64, depends on S3.C61, assured by S3.C65]  Implement an authorized default encryption key on each bucket; and enable S3 Bucket Key, if CloudTrail events are not required for KMS encrypt/decrypt (note: Amazon S3 evaluates and applies bucket policies before applying bucket default encryption settings) | Request 1) the mechanism implementing an authorized default encryption key on each bucket and enabling S3 Bucket Key, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Low | S3.FC1  S3.FC10  S3.FC5 | S3.T17 (Medium)  S3.T20 (High)  S3.T36 (High)  S3.T37 (High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C65]  Verify each bucket has an authorized default encryption key and has S3 Bucket Key enabled | Create/modify a bucket 1) without default encryption, 2) with a wrong default encryption key or 3) without S3 Bucket Key disabled, it should be detected. | Medium | S3.FC1  S3.FC10  S3.FC5 | - | Very High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C66, depends on S3.C61, assured by S3.C67]  Block PutObject requests with unauthorized KMS key on each bucket (e.g. using an S3 bucket policy deny statement on PutObject if the condition if exists "s3:x-amz-server-side-encryption-aws-kms-key-id" is not an authorized KMS key) | Make a request encrypted with an unauthorized KMS key, it should be denied | Low | S3.FC1  S3.FC10  S3.FC5 | S3.T11 (High)  S3.T16 (High)  S3.T17 (High)  S3.T20 (Very High)  S3.T30 (High)  S3.T36 (High)  S3.T37 (High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C67]  Verify all buckets block PutObject requests with an unauthorized KMS key (e.g. using the Config rule: [S3\_BUCKET\_POLICY\_NOT\_MORE\_PERMISSIVE](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-policy-not-more-permissive.html), note that a new rule needs be deployed for each configuration, then the resource tracked by name or tag; alternatively you might use [S3\_BUCKET\_SERVER\_SIDE\_ENCRYPTION\_ENABLED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-server-side-encryption-enabled.html) to ensure a limited coverage) | Create a bucket not blocking PutObject requests with an unauthorized KMS key, it should be detected. | Medium | S3.FC1  S3.FC10  S3.FC5 | - | Very High |
| Directive (COSO)  Identify (NIST CSF) | [S3.C145]  Maintain a list of buckets (or paths) required to be encrypted with server-side encryption with customer-provided keys (SSE-C) | Request the list of buckets (or paths) required to be encrypted with server-side encryption with customer-provided keys (SSE-C), its review process, and its review records | Medium | S3.FC10  S3.FC5 | S3.T11 (Very Low)  S3.T16 (Very Low)  S3.T20 (Very Low)  S3.T30 (Very Low)  S3.T36 (Very Low)  S3.T37 (Very Low) | Very High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C146, depends on S3.C145, assured by S3.C147]  For buckets (or paths) requiring SSE-C, block PutObject requests with unauthorized encryption (e.g. using an S3 bucket policy deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-customer-algorithm"="AES265" is not present) | Make a request to a bucket (or path) requiring SSE-C without the proper encryption, it should be denied | Low | S3.FC10  S3.FC5 | S3.T11 (High)  S3.T16 (High)  S3.T20 (Very High)  S3.T30 (High)  S3.T36 (High)  S3.T37 (High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C147]  For buckets (or paths) requiring SSE-C, verify all buckets block PutObject requests with unauthorized encryption | Create a bucket requiring SSE-C not blocking PutObject requests with unauthorized encryption, it should be detected. | High | S3.FC10  S3.FC5 | - | Very High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C152, assured by S3.C154]  Ensure bucket ACL and object ACL are disabled on each bucket | Request 1) the mechanism ensuring bucket ACL and object ACL are disabled on each bucket, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Medium | S3.FC1  S3.FC24  S3.FC25  S3.FC5  S3.FC8 | S3.T4 (Very High)  S3.T6 (Very High)  S3.T36 (Very High)  S3.T43 (Very High)  S3.T52 (Very High)  S3.T53 (Very High) | Very High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C153]  Prevent the creation of buckets with ACL enabled (e.g. by using a SCP and/or an IAM policy on "s3:CreateBucket" with a deny statement on StringNotEquals "s3:x-amz-object-ownership":"BucketOwnerEnforced"). Note that it does not block someone from enabling an ACL afterward via PutPutBucketOwnershipControls. | Create a bucket to enable ACL, it should be denied. | Low | S3.FC1  S3.FC24  S3.FC25  S3.FC5  S3.FC8 | S3.T4 (High)  S3.T6 (High)  S3.T36 (High)  S3.T43 (High)  S3.T52 (High)  S3.T53 (High) | Very High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C154]  Verify bucket ACL and object ACL are disabled on each bucket (e.g. using the AWS Config rule [S3\_BUCKET\_ACL\_PROHIBITED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-acl-prohibited.html) for bucket ACL) | Create/modify a bucket to enable ACL, it should be detected. | Medium | S3.FC1  S3.FC24  S3.FC25  S3.FC5  S3.FC8 | - | Very High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C1, depends on S3.C119, assured by S3.C2]  Block all unencrypted requests and unauthorized TLS version(s) from IAM entities you control (e.g. by denying all unencrypted requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" !=*authorized TLS version(s)*, using an SCP on your AWS Organization root node) | Make an unencrypted S3 API call, it should be denied. | Low | S3.FC1  S3.FC5 | S3.T12 (Very High)  S3.T34 (High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C2]  Verify the control blocking unencrypted requests and unauthorized TLS version(s) from IAM entities you control (e.g. an SCP on your AWS Organizations root node) is properly implemented | Remove the control blocking unencrypted requests and unauthorized TLS version(s) (e.g. the SCP on your root node), it should be detected. | High | S3.FC1  S3.FC5 | - | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C6, depends on S3.C119, assured by S3.C7]  Block all unencrypted requests to S3 bucket you control (e.g. by denying all requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" != *authorized TLS version(s)*, on the S3 bucket policy) | Make an unencrypted AWS API call to a bucket you control, it should be denied. | Low | S3.FC5 | S3.T34 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C7]  Verify all S3 bucket policies block unencrypted traffic (e.g. using the AWS Config rule: [S3\_BUCKET\_SSL\_REQUESTS\_ONLY](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-ssl-requests-only.html)) and unauthorized version(s) of TLS. | Remove the statement on a S3 bucket policy 1) denying all unencrypted requests and 2) denying unauthorized TLS versions, it should be detected. | Medium | S3.FC5 | - | High |
| Directive (COSO)  Identify (NIST CSF) | [S3.C119]  Maintain a list of authorized version(s) of TLS/SSL per bucket (or per account/OU/Org) | Request the list of authorized version(s) of TLS/SSL per bucket (or per account/OU/Org), its review mechanism and associated records | Low | S3.FC1 | S3.T12 (Very Low) | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C8]  Block S3 endpoints ([DNS](https://docs.aws.amazon.com/general/latest/gr/s3.html) and [IP ranges](https://aws.amazon.com/premiumsupport/knowledge-center/s3-find-ip-address-ranges/)) in your corporate perimeter security to the Internet (e.g. firewalls, or cloud interception proxy like [Kivera](https://kivera.io)) including via Internet Gateway, to force usage of VPC endpoints. It will block data-plane transfer. Note: AWS console stays functional as it proxies non-data-plane requests (via "console.aws.amazon.com"). | Request the evidence of the implementation of blocking S3 endpoints in your corporate perimeter security (e.g. firewalls) and tests of its effectiveness. | Low | S3.FC1  S3.FC28  S3.FC5  S3.FC7 | S3.T7 (High)  S3.T10 (High)  S3.T12 (Low)  S3.T18 (Medium)  S3.T34 (Very High) | High |
| Directive (COSO)  Detect (NIST CSF) | [S3.C118]  Enable [S3 policy findings in Amazon Macie](https://docs.aws.amazon.com/macie/latest/user/findings-types.html#findings-policy-types) in all AWS accounts in all Regions, and protect it using Macie ThreatModel | Request the Macie ThreatModel and the evidence of its application for enabling and protecting S3 policy findings | Very Low | S3.FC10  S3.FC15  S3.FC5  S3.FC8 | S3.T2 (Medium)  S3.T4 (Medium)  S3.T22 (Medium)  S3.T36 (Medium)  S3.T37 (Medium)  S3.T38 (Medium) | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C15]  Request access via an S3 access point on a bucket you don't own, if compatible with your interaction with the bucket (e.g. not through not-compatible AWS service) | Request the documented reason(s) access point was not implemented in the use case | Low | S3.FC1 | S3.T8 (Medium)  S3.T9 (Medium)  S3.T31 (Very High) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C114, depends on S3.C11]  For all external buckets with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover, block the PutObject with any ACL (e.g. using IAM or SCP and a deny on the condition "StringLike": {"s3:x-amz-acl": "\*"}). It should be called via PutObjectAcl. | Make a request to an external bucket with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover requirement, it should be denied. | High | S3.FC1 | S3.T43 (Very High) | High |
| Directive (COSO)  Identify (NIST CSF) | [S3.C17]  For each VPC, maintain a list of AWS Organizations, OU, and/or AWS account(s) where IAM entities are authorized to access S3 | For each VPC, request the list of AWS Organizations, OU, and/or AWS account(s), where IAM entities are authorized to access S3, its review process, and its review records | Medium | S3.FC1  S3.FC5 | S3.T9 (Very Low)  S3.T11 (Very Low) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C19, depends on S3.C17, assured by S3.C20]  Block any IAM entity not belonging to an authorized AWS Organizations, OU, and/or AWS account(s) to call S3 from your VPCs by adding a deny statement on the S3 VPC endpoint policy of each VPC, with the condition using "aws:PrincipalOrgPaths" ([ref](https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_condition-keys.html#condition-keys-principalorgpaths)) including the full Org ID, as those are globally unique. | For each VPC, do an API call with an IAM entity which is not part of its authorized AWS Organizations path(s), it should be denied. | Low | S3.FC1  S3.FC5 | S3.T9 (Very High)  S3.T11 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C20]  Verify all S3 VPC endpoint are blocking any IAM entity not belonging to an authorized AWS Organizations, OU and/or AWS account(s) | Remove the policy statement blocking any IAM entity not belonging to an authorized AWS Organizations, OU and/or AWS account(s) from the VPC endpoint, it should be detected. | High | S3.FC1  S3.FC5 | - | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C26]  Limit the access to the IAM actions required to execute the threats using AWS IAM and/or SCP, following the IAM Operating Model and using the IAM ThreatModel. Use the [IAM Access Advisor](https://aws.amazon.com/blogs/security/tighten-s3-permissions-iam-users-and-roles-using-access-history-s3-actions/) to review the usage of non-object-related S3 actions. | Request the list of authorized IAM principals that have the permissions required to execute the threat actions, its review process, and its review records | High | S3.FC1  S3.FC10  S3.FC12  S3.FC13  S3.FC15  S3.FC19  S3.FC2  S3.FC20  S3.FC24  S3.FC25  S3.FC26  S3.FC27  S3.FC32  S3.FC33  S3.FC5  S3.FC6  S3.FC7  S3.FC8 | S3.T1 (High)  S3.T2 (High)  S3.T5 (Low)  S3.T6 (Medium)  S3.T7 (High)  S3.T8 (High)  S3.T11 (High)  S3.T14 (Very High)  S3.T16 (High)  S3.T17 (Very High)  S3.T18 (High)  S3.T21 (Medium)  S3.T25 (High)  S3.T26 (High)  S3.T28 (High)  S3.T30 (High)  S3.T33 (Very High)  S3.T35 (Very High)  S3.T36 (Medium)  S3.T37 (Very High)  S3.T38 (Very High)  S3.T39 (High)  S3.T41 (High)  S3.T42 (High)  S3.T44 (High)  S3.T46 (High)  S3.T47 (High)  S3.T48 (High)  S3.T49 (High)  S3.T50 (High)  S3.T51 (High)  S3.T52 (High)  S3.T53 (High)  S3.T54 (High)  S3.T55 (High)  S3.T56 (High)  S3.T58 (High)  S3.T59 (High) | High |
| Directive (COSO)  Identify (NIST CSF) | [S3.C31]  Maintain a list of authorized AWS accounts to provide KMS keys for S3 for each AWS account | Request the list of authorized AWS accounts to provide KMS keys for S3 for each AWS account, its review process, and its review records | Medium | S3.FC1  S3.FC15  S3.FC26  S3.FC5  S3.FC8 | S3.T1 (Very Low)  S3.T2 (Very Low)  S3.T4 (Very Low)  S3.T5 (Very Low)  S3.T7 (Very Low)  S3.T8 (Very Low)  S3.T9 (Very Low)  S3.T11 (Very Low)  S3.T16 (Very Low)  S3.T21 (Very Low)  S3.T27 (Very Low)  S3.T28 (Very Low)  S3.T30 (Very Low)  S3.T31 (Very Low) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C32, depends on S3.C31]  Block requests with unauthorized AWS account providing the KMS key (e.g. using an SCP, bucket policy, and VPC endpoint deny statement on PutObject if the condition "s3:x-amz-server-side-encryption-aws-kms-key-id" is not a KMS key from an authorized AWS account) | Make a request encrypted with a KMS key from unauthorized AWS account, it should be denied | Low | S3.FC1  S3.FC15  S3.FC26  S3.FC5  S3.FC8 | S3.T1 (High)  S3.T2 (Medium)  S3.T4 (High)  S3.T5 (High)  S3.T7 (High)  S3.T8 (High)  S3.T9 (High)  S3.T11 (Medium)  S3.T16 (High)  S3.T21 (Medium)  S3.T27 (Low)  S3.T28 (High)  S3.T30 (High)  S3.T31 (High) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C34, assured by S3.C36]  Deny requests to change object ACL to public (e.g. using an SCP, S3 bucket policy, and VPC endpoint policy blocking PutObjectAcl for "s3:x-amz-grant-read", "s3:x-amz-grant-read-acp", "s3:x-amz-grant-write-acp", "s3:x-amz-grant-full-control" on the following predefined groups "http://acs.amazonaws.com/groups/global/AllUsers" and "http://acs.amazonaws.com/groups/global/AuthenticatedUsers") | Make a call to create a public ObjectACL, it should be denied. | Medium | S3.FC1  S3.FC5 | S3.T6 (Very High)  S3.T36 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C36]  Verify the control blocking change ObjectACL to public (e.g. an SCP and VPC endpoint policy) is properly implemented | Remove the control blocking changes of ObjectACL to public, it should be detected. | High | S3.FC1  S3.FC5 | - | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C38, assured by S3.C39]  Block the action "s3:DeleteBucket" (e.g. via SCP, exemption can be managed by authorizing a SuperAdmin to delete buckets with a certain tag, and with bucket owners able to tag bucket) | Do a DeleteBucket, it should be denied | Low | S3.FC5 | S3.T1 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C39]  Verify the control blocking the action "s3:DeleteBucket" (e.g. an SCP on your AWS Organizations root node) is properly implemented | Remove the control blocking the action "s3:DeleteBucket" (e.g. an SCP on your root node), it should be detected. | High | S3.FC5 | - | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C113, depends on S3.C11]  When transmitting an object to an external bucket with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover, use 2 separate APIs (PutObject and PutObjectAcl) instead of the built-in object ACL operation in PutObject. | Request the process on ensuring that PutObject requests on external bucket with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover use 2 separate APIs | Medium | S3.FC1 | S3.T43 (High) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C55, assured by S3.C57]  Deny requests to add bucket ACL (e.g. using an SCP, bucket policy, and VPC endpoint policy blocking "s3:PutBucketAcl") | Make a call to create a bucket ACL, it should be denied. | Medium | S3.FC19  S3.FC8 | S3.T4 (Very High)  S3.T58 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C57]  Verify the control blocking bucket ACL changes (e.g. an SCP, a bucket policy and VPC endpoint policy) is properly implemented | Remove the control blocking bucket ACL changes, it should be detected. | High | S3.FC19  S3.FC8 | - | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C58]  Track all buckets you control, define their authorized data classification, identify whether the hosted data is primary (i.e. source of truth, for example logs, backups, forensic data, raw data, etc.) or an input/output of a process (e.g. file-processing, software package, etc.), their WORM requirements (e.g. SEC 17a-4, CTCC, etc.), if they are production/non-production (preferably done at account-level), their storage class. You may use tags, Infra-as-code, AWS Glue Data Catalog, or external management tools like [FINRA herd](https://finraos.github.io/herd/)) | Request the list of all buckets you control define their authorized data classification, and identify whether the data is primary and the mechanism and records to ensure the accuracy of those metadata | High | S3.FC1  S3.FC13  S3.FC5 | S3.T11 (High)  S3.T17 (Very High)  S3.T25 (Low) | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C69, depends on S3.C58, assured by S3.C70]  Enable versioning on buckets holding primary data | Request the mechanism used to ensure versioning on buckets holding primary data, and its records | Very Low | S3.FC1  S3.FC5 | S3.T16 (High)  S3.T17 (High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C70]  Verify buckets holding primary data are versioned (e.g. using [S3\_BUCKET\_VERSIONING\_ENABLED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-versioning-enabled.html)) | Remove versioning from a bucket holding primary data, it should be detected | Low | S3.FC1  S3.FC5 | - | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C74, depends on S3.C58, assured by S3.C75]  Implement the authorized default S3 Object Lock on each bucket (note: Amazon S3 evaluates and applies bucket policies before applying bucket default S3 Object Lock settings) | Upload an object without appropriate S3 Object Lock, it should have the S3 Object Lock automatically. | Low | S3.FC1  S3.FC13  S3.FC5 | S3.T16 (High)  S3.T17 (High)  S3.T25 (High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C75]  Verify all buckets have the correct default S3 Object Lock configuration | Create a bucket 1) without S3 Object Lock or 2) with an incorrect default S3 Object Lock, it should be detected. | Medium | S3.FC1  S3.FC13  S3.FC5 | - | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C76, depends on S3.C58, assured by S3.C77]  Block PutObject and PutObjectRetention requests with unauthorized S3 Object Lock on each bucket (e.g. using an S3 bucket policy deny statement on PutObject and PutObjectRetention if the condition if exists "s3:object-lock-mode" and "s3:object-lock-remaining-retention-days" is not the defined S3 Object Lock configuration) | Make a request with an incorrect S3 Object Lock configuration, it should be denied | Low | S3.FC1  S3.FC5 | S3.T16 (High)  S3.T17 (High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C77]  Verify all buckets blocks PutObject and PutObjectRetention requests with unauthorized S3 Object Lock (e.g. using the Config rule: [S3\_BUCKET\_POLICY\_NOT\_MORE\_PERMISSIVE](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-policy-not-more-permissive.html), note that a new rule needs be deployed for each configuration, then the resource tracked by name or tag) | Create a bucket not blocking PutObject and PutObjectRetention requests with unauthorized S3 Object Lock, it should be detected. | Medium | S3.FC1  S3.FC5 | - | High |
| Directive (COSO)  Identify (NIST CSF) | [S3.C98]  For each S3 bucket, maintain a list of VPC(s) authorized to access it. | For each S3 bucket, request the list of authorized VPC to access it, its review process, and its review records | Low | S3.FC1  S3.FC10  S3.FC2  S3.FC5 | S3.T11 (Very Low)  S3.T14 (Very Low)  S3.T17 (Very Low)  S3.T30 (Very Low)  S3.T33 (Very Low)  S3.T36 (Very Low)  S3.T38 (Very Low)  S3.T39 (Very Low) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C99, depends on S3.C98, assured by S3.C100]  Limit the access to only those VPC(s) (e.g. using S3 bucket statement, deny if the condition "aws:SourceVpce", or if the bucket policy size is beyond the limit, use this condition on access point) | Make a request to the bucket outside an authorized VPC, it should be denied | Very Low | S3.FC1  S3.FC10  S3.FC2  S3.FC5 | S3.T11 (Medium)  S3.T14 (Medium)  S3.T17 (Medium)  S3.T30 (High)  S3.T33 (High)  S3.T36 (High)  S3.T38 (High)  S3.T39 (High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C100]  Verify all buckets include a control to limit access to only authorized VPC(s) (e.g. using the AWS Config rule [S3\_BUCKET\_POLICY\_GRANTEE\_CHECK](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-policy-grantee-check.html)) | Remove the control limiting access to only authorized VPC(s), it should be detected. | Medium | S3.FC1  S3.FC10  S3.FC2  S3.FC5 | - | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C105, depends on S3.C104, assured by S3.C109]  Limit access via the S3 access point by using a VPC endpoint and/or bucket policy with the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" in an allow statement to reduce the length of the allowlist bucket name in the VPC endpoint/bucket policy. | Do a request on an unauthorized access point or bucket, it should be denied. | Medium | S3.FC1  S3.FC26  S3.FC28  S3.FC33  S3.FC5 | S3.T7 (Medium)  S3.T9 (Very High)  S3.T10 (Very High)  S3.T11 (Medium)  S3.T54 (Medium)  S3.T55 (Medium) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C106, assured by S3.C110]  In the S3 bucket policy, deny all IAM principals not using an authorized S3 access point(s) using the condition "s3:DataAccessPointAccount" or preferably "s3:DataAccessPointArn" | Query the bucket outside S3 access point, it should be denied. | Medium | S3.FC1  S3.FC26  S3.FC33 | S3.T7 (Medium)  S3.T28 (High)  S3.T55 (Medium)  S3.T56 (Very High) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C107]  Block the creation "s3:CreateAccessPoint" of non-VPC S3 access point (e.g. using the condition "StringNotEquals": {"s3:AccessPointNetworkOrigin": "VPC"}) | Do a request to create an internet-based access point, it should be denied. | Low | S3.FC1  S3.FC26 | S3.T7 (Medium)  S3.T28 (Very High) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C108, assured by S3.C111]  Block all traffic from Internet-configured S3 access point (e.g. on the bucket policy, using a deny statement with the condition "StringNotEquals": {"s3:AccessPointNetworkOrigin": "VPC"}) | Create an internet-facing access point and try to access a bucket, it should be denied. | Low | S3.FC1  S3.FC26  S3.FC28 | S3.T7 (Medium)  S3.T10 (Medium)  S3.T28 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C109]  Verify only access points are used in the resource-level statement in VPC endpoints | Create a VPC endpoint giving access to an S3 bucket, it should be detected. | High | S3.FC1  S3.FC26  S3.FC28  S3.FC33  S3.FC5 | - | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C110]  Verify S3 bucket policies deny non-authorized S3 access points | Remove/modify the deny on the bucket policy, it should be detected. | High | S3.FC1  S3.FC26  S3.FC33 | - | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C111]  Verify all S3 access points are VPC attached | Create an internet-based access point, it should be detected. | Low | S3.FC1  S3.FC26  S3.FC28 | - | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C112, depends on S3.C104]  Block any [object-related operations](https://docs.aws.amazon.com/AmazonS3/latest/dev/using-access-points.html#access-points-service-api-support) access to S3 bucket not through access point (i.e. using a deny IAM policy statement with the condition "ArnNotLike" {"s3:DataAccessPointArn": "arn:aws:s3:*Region*:*AccountId*:accesspoint/\*"}) | Access any S3 bucket using the S3 public endpoint, it should be denied. | Low | S3.FC1  S3.FC5 | S3.T7 (Medium)  S3.T11 (High) | High |
| Directive (COSO)  Identify (NIST CSF) | [S3.C120]  Maintain a list of IAM roles used for Batch job, ideally dedicated (e.g. using change management process on infrastructure-as-code) | Request the list of all IAM roles configured for Batch job | Medium | S3.FC27 | S3.T44 (Very Low) | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C122, depends on S3.C120]  Limit access to authorized IAM roles used for Batch job, using the IAM ThreatModel (e.g. trust policy, and "iam:PassRole") | Request the IAM ThreatModel and the evidence of its application to the IAM roles used for Batch job | Medium | S3.FC27 | S3.T44 (Very High) | High |
| Directive (COSO)  Identify (NIST CSF) | [S3.C125]  Maintain a list of authorized Lambda function for each Object Lambda access point, its associated access point, its associated GET request(s), and payload | Request the list of authorized Lambda function for each Object Lambda access point, its associated access point, its associated GET request(s), and payload, its review process, and its review records | Low | S3.FC32 | S3.T46 (Very Low) | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C126, depends on S3.C125, assured by S3.C127]  Ensure only authorized Lambda function for each Object Lambda access point, its associated access point, its associated GET request(s), and payload are created | Request the mechanism ensuring only authorized Lambda function for each Object Lambda access point, its associated access point, its associated GET request(s), and payload, and the evidence of its execution | Medium | S3.FC32 | S3.T46 (Very High) | High |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C127]  Verify only the authorized Lambda function are configured on each Object Lambda access point, its associated access point, its associated GET request(s), and payload | Attach 1) an unauthorized Lambda function on an Object Lambda access point, 2) an unauthorized Object Lambda access point to an access point, 3) an authorized Lambda function with an unauthorized GET request on an Object Lambda access point, and 4) an authorized Lambda function with an unauthorized payload, it should be detected. | Medium | S3.FC32 | - | High |
| Directive (COSO)  Protect (NIST CSF) | [S3.C144, depends on S3.C141]  Ensure S3 website endpoints are protected with HTTP headers ([ref](https://aws.amazon.com/blogs/networking-and-content-delivery/adding-http-security-headers-using-lambdaedge-and-amazon-cloudfront/)) using a CDN (e.g. CloudFront) | Request the mechanism ensuring S3 website endpoints are protected with HTTP headers | Medium | S3.FC16 | S3.T13 (High)  S3.T29 (Very High) | High |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C3, depends on S3.C119, assured by S3.C5]  Block all unencrypted requests and unauthorized TLS version(s) from VPC endpoints you control (e.g. by denying all requests with the condition "aws:SecureTransport" = False, or by using "s3:TlsVersion" != *authorized TLS version(s)*, on the VPC endpoint policy) | Make an unencrypted AWS API call from one of your VPCs with VPC endpoint, it should be denied. | Low | S3.FC1  S3.FC5 | S3.T12 (Medium)  S3.T34 (Medium) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C5]  Verify a statement exists on all your VPC endpoint policy denying all requests with the condition "aws:SecureTransport" = False | Create/remove the statement on a VPC endpoint policy denying 1) all unencrypted requests or 2) unauthorized TLS version(s), it should be detected. | High | S3.FC1  S3.FC5 | - | Medium |
| Directive (COSO)  Detect (NIST CSF) | [S3.C9]  Enable [CloudTrail S3 data events](https://docs.aws.amazon.com/awscloudtrail/latest/userguide/logging-data-events-with-cloudtrail.html#logging-data-events) in relevant AWS accounts, Regions, and buckets (e.g. production, with sensitive data, etc.). Make it available for security analysis, and protect it using CloudTrail ThreatModel | Request the CloudTrail ThreatModel and the evidence of its application for enabling and protecting S3 data events | Very Low | S3.FC1  S3.FC5  S3.FC8 | S3.T1 (Low)  S3.T4 (Low)  S3.T5 (Low)  S3.T6 (Low)  S3.T7 (Low)  S3.T8 (Low)  S3.T9 (Low)  S3.T11 (Low)  S3.T12 (Low)  S3.T16 (Low)  S3.T21 (Low)  S3.T31 (Low)  S3.T34 (Low)  S3.T35 (Low)  S3.T36 (Low)  S3.T39 (Low) | Medium |
| Directive (COSO)  Detect (NIST CSF) | [S3.C10]  Enable and monitor [S3 protection in Amazon GuardDuty](https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_finding-types-s3.html) in all AWS accounts in all Regions, and protect it using GuardDuty ThreatModel. Ensure findings are investigated (e.g. using Amazon Detective). | Request the GuardDuty ThreatModel and the evidence of its application for enabling, monitoring, investigation and protecting S3 protection | Low | S3.FC1  S3.FC24  S3.FC25  S3.FC5  S3.FC8 | S3.T3 (Low)  S3.T4 (Low)  S3.T16 (Medium)  S3.T52 (Medium)  S3.T53 (Medium) | Medium |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C12, depends on S3.C11]  Allow only authorized ACL on objects for buckets you don't control (e.g. using IAM and VPC endpoint policy with the [ACL conditions](https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#acl-specific-condition-keys)) | Put an object with an unauthorized ACL, it should be denied. | Medium | S3.FC1 | S3.T5 (Medium)  S3.T6 (High) | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C13, depends on S3.C11]  Monitor that only authorized external buckets are used (e.g. via CloudTrail S3 data events in resources[].accountId and resources[].ARN). Both account ID and bucket name must be verified. | Make a call to an unauthorized bucket, it should be detected | Low | S3.FC1  S3.FC5 | S3.T1 (Low)  S3.T7 (Low)  S3.T11 (Low)  S3.T21 (Low)  S3.T31 (Medium) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C14, depends on S3.C11]  Scan all data before uploading to an external bucket to ensure the classification of the data is aligned with the bucket classification (e.g. using Macie). | Request 1) the mechanism ensuring all data are scanned for proper data classification before upload to an external bucket are configured, 2) its records of execution for all object upload flows, and 3) plan to move any older object upload flows | High | S3.FC1  S3.FC16  S3.FC5 | S3.T5 (High)  S3.T14 (High)  S3.T15 (Medium) | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C115, depends on S3.C11]  For all external bucket with bucket-owner-full-control ACL requirements but without S3 Object Ownership handover, monitor that the PutObject do not include the ACL operation | Make a request to an external bucket with bucket-owner-full-control ACL requirement but without S3 Object Ownership handover requirement, it should be detected. | Low | S3.FC1 | S3.T43 (Low) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C16]  Analyze and protect all AWS services accessing S3 (e.g. via ThreatModel). Enforce usage in VPC only, whenever possible. | Request the threat and mitigating controls for all AWS services using S3. | High | S3.FC1  S3.FC5 | S3.T21 (Very High)  S3.T30 (Very High) | Medium |
| Directive (COSO)  Identify (NIST CSF) | [S3.C22]  Maintain a list of authorized S3 and S3 access point (and their respective AWS accounts) to be accessed for each VPC | Request the list of authorized S3 and S3 access point to be access for each VPC, its review process, and its review records | Medium | S3.FC1  S3.FC5 | S3.T8 (Very Low)  S3.T9 (Very Low)  S3.T11 (Very Low) | Medium |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C23, depends on S3.C22, assured by S3.C24]  Limit the access to only authorized S3 bucket(s) or their AWS account(s) from each VPC (e.g. using the condition key "s3:ResourceAccount" on the VPC endpoint policy, alternatively use a specific resource-level statement for each bucket, or if the VPC endpoint policy size is beyond the limit and more granular control on VPC is required, use access points) | Make a request to an unauthorized bucket from one of your VPC, it should be denied | Medium | S3.FC1  S3.FC5 | S3.T8 (High)  S3.T9 (High)  S3.T11 (High) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C24]  Verify all VPC are limited to limit access to only authorized S3 bucket(s) | Remove the control limiting access to only authorized S3 bucket(s), it should be detected. | High | S3.FC1  S3.FC5 | - | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C27, assured by S3.C28]  In the S3 bucket/access point/Object Lambda access point policy, do not allow IAM principals of the same AWS account. Only AWS IAM should be used to provide permissions to a principal of the same AWS account. | Request all S3 bucket/access point/Object Lambda access point policy statements with "allow", no principal from the same account should be authorized. | Low | S3.FC1  S3.FC10  S3.FC12  S3.FC13  S3.FC15  S3.FC19  S3.FC2  S3.FC20  S3.FC26  S3.FC27  S3.FC32  S3.FC33  S3.FC5  S3.FC7 | S3.T1 (Low)  S3.T2 (Low)  S3.T6 (Low)  S3.T7 (Low)  S3.T8 (Low)  S3.T11 (Low)  S3.T14 (Medium)  S3.T16 (Low)  S3.T17 (Medium)  S3.T18 (Low)  S3.T21 (Low)  S3.T25 (Low)  S3.T26 (Medium)  S3.T30 (Low)  S3.T33 (Medium)  S3.T35 (Medium)  S3.T36 (Low)  S3.T37 (Medium)  S3.T38 (Medium)  S3.T39 (Low)  S3.T41 (Low)  S3.T42 (Low)  S3.T44 (Low)  S3.T46 (Medium)  S3.T54 (Medium)  S3.T55 (Medium)  S3.T58 (Medium)  S3.T59 (Medium) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C28]  Verify all S3 bucket/access point/Object Lambda access point policies do not allow an IAM principal of the same AWS account (e.g. using the Config rule S3\_BUCKET\_POLICY\_GRANTEE\_CHECK for bucket policy) | Add an allow statement for an IAM principal of the same account in 1) a bucket policy, 2) in an access point policy, and 3) in an Object Lambda access point, it should be detected. | Medium | S3.FC1  S3.FC10  S3.FC12  S3.FC13  S3.FC15  S3.FC19  S3.FC2  S3.FC20  S3.FC26  S3.FC27  S3.FC32  S3.FC33  S3.FC5  S3.FC7 | - | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C44]  Use "x-amz-checksum" from the [object metadata](https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingMetadata.html#SysMetadata) to validate the integrity of the object instead of etag. If etag is used, make sure properly account for its different definitions ([ref](https://teppen.io/2018/06/23/aws_s3_etags/)) | Request 1) the mechanism ensuring checksum are being used instead of etag, and otherwise ensuring etag different definitions are properly accounted for, and 2) plan to move any older system using etag to use the checksum metadata. | Low | S3.FC1  S3.FC5 | S3.T17 (Medium)  S3.T27 (High) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C46]  Ensure all S3 buckets interacted with are in the correct AWS account (e.g. using the condition in all compatible S3 requests: x-amz-expected-bucket-owner and x-amz-source-expected-bucket-owner) | Request the process on ensuring that all S3 buckets interacted with are in the correct AWS account | Medium | S3.FC1  S3.FC5 | S3.T1 (Medium)  S3.T3 (Medium) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C47, assured by S3.C48]  Front buckets that are required to be public, using authenticated CDN (e.g. CloudFront) or API Gateway, protected with WAF (e.g. for [hotlinking](https://aws.amazon.com/blogs/security/how-to-prevent-hotlinking-by-using-aws-waf-amazon-cloudfront-and-referer-checking/)) | Request the process ensuring that buckets required to be public are front by authenticated CDN or API Gateway | Medium | S3.FC16  S3.FC5 | S3.T13 (Very High)  S3.T14 (Medium)  S3.T22 (Very High) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C48]  Verify no bucket is available publicly for write or read (e.g. using the AWS Config rules: [S3\_BUCKET\_PUBLIC\_READ\_PROHIBITED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-public-read-prohibited.html) and [S3\_BUCKET\_PUBLIC\_WRITE\_PROHIBITED](https://docs.aws.amazon.com/config/latest/developerguide/s3-bucket-public-write-prohibited.html)) | Create a public S3 bucket, it should be detected. | Very Low | S3.FC16  S3.FC5 | - | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C59, depends on S3.C58]  Use a data discovery tool (e.g. Amazon Macie) to ensure no sensitive data is stored in an unauthorized bucket | Upload a higher classification data in a bucket, it should be detected. | Medium | S3.FC5 | S3.T11 (Medium) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C62]  Verify all objects on S3 buckets are encrypted with an authorized KMS key (e.g. using S3 inventory, see [blog](https://aws.amazon.com/blogs/storage/encrypting-objects-with-amazon-s3-batch-operations/), or [S3 Storage Lens](https://docs.aws.amazon.com/AmazonS3/latest/dev/storage_lens_basics_metrics_recommendations.html#storage_lens_basics_metrics_types)) | Upload an encrypted data using an unauthorized KMS key, it should be detected. | Medium | S3.FC1  S3.FC10  S3.FC5 | - | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C140, assured by S3.C62]  Ensure all objects on S3 buckets are encrypted with an authorized KMS key | Request the mechanism (including training, or utility) ensuring only authorized KMS key are used for any objects stored in S3 | Medium | S3.FC1  S3.FC10  S3.FC5 | S3.T11 (Medium)  S3.T16 (Medium)  S3.T17 (Medium)  S3.T20 (Medium)  S3.T30 (Medium)  S3.T36 (Medium)  S3.T37 (Medium) | Medium |
| Directive (COSO)  Recover (NIST CSF) | [S3.C71, depends on S3.C58]  Backup primary data in a secure location under a different security authority (e.g. in an [AWS data bunker account](https://wellarchitectedlabs.com/security/100_labs/100_create_a_data_bunker/1_instructions/) via replication, or using AWS Backup for Amazon S3) | Request the mechanism used to backup primary data in a location which have different security authority, its records of execution, and records of restoration testing | Medium | S3.FC1  S3.FC13  S3.FC5 | S3.T16 (High)  S3.T17 (High)  S3.T25 (High) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C72]  Aligned with your data governance, encrypt on the client side - or tokenize - appropriate data | Request the governance and mechanism(s) used to protect data (e.g. encrypt or tokenize critical data on the client side) | Very High | S3.FC1  S3.FC10  S3.FC16  S3.FC5 | S3.T1 (Medium)  S3.T3 (Medium)  S3.T5 (High)  S3.T7 (High)  S3.T11 (Very High)  S3.T12 (Very High)  S3.T13 (Very High)  S3.T17 (High)  S3.T20 (High)  S3.T30 (High)  S3.T31 (High) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C73]  Create a process to apply a legal hold to any S3 bucket whenever required. The condition "s3:object-lock-legal-hold" can be used to restrict who can remove such a lock. | Request the process of applying legal hold, and its records | Medium | S3.FC1  S3.FC5 | S3.T16 (Low)  S3.T17 (Medium) | Medium |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C84]  Block all requests not using HTTP authorization header, i.e. presign via query strings or POST ([ref](https://docs.aws.amazon.com/AmazonS3/latest/API/sig-v4-authenticating-requests.html)) (e.g. using an SCP and S3 policy on all buckets with deny on "StringNotEquals":{"s3:authType": "REST-HEADER"}). Note that it blocks uploads via the console, as well. | Make a request with a non-HTTP authorization header, it should be denied | Low | S3.FC5 | S3.T39 (Medium) | Medium |
| Directive (COSO)  Identify (NIST CSF) | [S3.C86]  Maintain a list of authorized buckets to have replication enabled, their target bucket and replication type (i.e. ownership, RTC, etc.) ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/replication-add-config.html)). | Request the list of authorized buckets to have replication enabled, their target bucket and replication rights, its review process, and its review records | Medium | S3.FC15 | S3.T2 (Very Low) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C87]  Verify only authorized buckets have replication enabled and with correct configuration | Configure replication on a non-authorized bucket, it should be detected | Medium | S3.FC15 | - | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C88]  Verify authorized buckets have the correct replication configuration | Modify the configuration of an authorized replication, it should be detected | High | S3.FC15 | - | Medium |
| Directive (COSO)  Identify (NIST CSF) | [S3.C89]  Maintain a list of IAM roles used for replication, ideally dedicated (e.g. using change management process on infrastructure-as-code) | Request the list of all IAM roles configured for replication | Medium | S3.FC15 | S3.T2 (Very Low) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C90, depends on S3.C89]  Limit the S3 access to the source/destination bucket and replication rights of each authorized IAM role configured for replication | Request the S3 access of replication role, and how they aligned to the replication requirements | Medium | S3.FC15 | S3.T2 (Medium) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C91, depends on S3.C89]  Limit access to authorized IAM roles used for replication, using the IAM ThreatModel (e.g. trust policy, and "iam:PassRole") | Request the IAM ThreatModel and the evidence of its application to the IAM roles used for replication | High | S3.FC15 | S3.T2 (High) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C92]  Verify only the authorized IAM role is configured for each replication | Create/modify a replication with an unauthorized IAM role, it should be detected | High | S3.FC15 | - | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C117]  Verify all replicated buckets have metrics enabled on each replication rule (included by default in S3 RTC) | Modify the replication metric of an authorized replication, it should be detected | Medium | S3.FC15 | - | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C134, depends on S3.C86, assured by S3.C87,S3.C88,S3.C117]  Ensure only authorized buckets have replication enabled and with correct configuration are configured | Request 1) the mechanism ensuring only authorized buckets have replication enabled and with correct configuration are configured, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Medium | S3.FC15 | S3.T2 (High) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C138, depends on S3.C89, assured by S3.C92]  Ensure only authorized IAM roles are attached for each replication, ideally dedicated | Request the mechanism ensuring authorized IAM roles are attached for each replication, and the evidence of its execution for all replication configuration | Medium | S3.FC15 | S3.T2 (High) | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C93, depends on S3.C58]  If the bucket is used as an input or the output of a process, scan the objects for malware (e.g. using [VirusScan](https://s3-virusscan.widdix.net/), [Cloud Storage Security](https://cloudstoragesec.com/amazon-s3), or [Trend Micro Cloud One](https://aws.amazon.com/blogs/apn/amazon-s3-malware-scanning-using-trend-micro-cloud-one-and-aws-security-hub/)) | Inject a malware test file, it should be detected. | Medium | S3.FC16  S3.FC5 | S3.T14 (Medium)  S3.T15 (Low) | Medium |
| Directive (COSO)  Identify (NIST CSF) | [S3.C101]  Maintain a list of authorized CloudFront distribution (via Origin Access Control) and associated bucket | Request the list of all authorized CloudFront distribution and associated S3 buckets | Low | S3.FC10 | S3.T20 (Very Low) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C102]  Verify all associations of CloudFront distributions with buckets are authorized | Create a non-authorized distribution or association, it should be detected. | High | S3.FC10 | - | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C137, depends on S3.C101, assured by S3.C102]  Ensure only authorized CloudFront distributions are associated with their authorized bucket, and vice versa. | Request 1) the mechanism ensuring only authorized only authorized CloudFront distribution are associated with their authorized bucket, and vice versa, 2) its records of execution for all new CloudFront distribution, and 3) plan to move any older CloudFront distribution | Medium | S3.FC10 | S3.T20 (High) | Medium |
| Directive (COSO)  Identify (NIST CSF) | [S3.C104]  Maintain a list of authorized access between VPCs, S3 access points, and S3. | Request the list of authorized access between VPC and S3 access points | Medium | S3.FC1  S3.FC26  S3.FC28  S3.FC5 | S3.T7 (Very Low)  S3.T9 (Very Low)  S3.T10 (Very Low)  S3.T11 (Very Low)  S3.T28 (Very Low) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C121, depends on S3.C120]  Limit the access to only required resources/permissions (e.g. source/destination bucket, Lambda functions) of each authorized IAM role configured for Batch jobs | Request the access to only required resources/permissions for each Batch IAM role, and how they aligned to the replication requirements | Medium | S3.FC27 | S3.T44 (High) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C123]  Verify only the authorized IAM role is configured for each Batch job | Create/modify a Batch job with an unauthorized IAM role, it should be detected | High | S3.FC27 | - | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C139, depends on S3.C120, assured by S3.C123]  Ensure only an authorized IAM role is attached on each Batch job | Request the mechanism ensuring only an authorized IAM role is attached on each Batch job, and the evidence of its execution for all new {resource} | Medium | S3.FC27 | S3.T44 (High) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C128]  Ensure Lambda functions configured on Object Lambda access point are secured using Lambda ThreatModel | Request the mechanism ensuring Lambda ThreatModel and its application for Lambda functions associated to Object Lambda access point, and its records of execution | Medium | S3.FC32 | S3.T46 (High) | Medium |
| Directive (COSO)  Identify (NIST CSF) | [S3.C129]  Maintain a list of cross-account access on each Object Lambda access point | Request the list of authorized cross-account access for each Object Lambda access point, its review process, and its review records | Very Low | S3.FC32 | S3.T46 (Very Low) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C130, depends on S3.C129, assured by S3.C131]  Ensure only authorized cross-account IAM entities are allowed in the Object Lambda access point policy | Request the mechanism ensuring only cross-account IAM entities are allowed in the Object Lambda access point policy, and the evidence of its execution | Low | S3.FC32 | S3.T46 (Medium) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C131]  Verify only the authorized cross-account IAM entities are allowed in the Object Lambda access point policy | Add 1) an unauthorized cross-account IAM entity on an Object Lambda access point policy, it should be detected. | High | S3.FC32 | - | Medium |
| Directive (COSO)  Identify (NIST CSF) | [S3.C141]  Maintain a list of authorized buckets to be configured as a S3 website endpoint | Request the list of authorized buckets to be configured as a website endpoint, its review process, and its review records | Low | S3.FC16 | S3.T13 (Very Low)  S3.T29 (Very Low) | Medium |
| Directive (COSO)  Protect (NIST CSF) | [S3.C142, depends on S3.C141, assured by S3.C143]  Ensure only authorized buckets are configured as a S3 website endpoint | Request 1) the mechanism ensuring only authorized buckets are configured as a S3 website endpoint, 2) its records of execution for all new website-enabled buckets, and 3) plan to move any older website-enabled buckets | Medium | S3.FC16 | S3.T13 (Medium)  S3.T29 (Medium) | Medium |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C143]  Verify only authorized buckets are configured as S3 website endpoints | Enable static website hosting on an unauthorized bucket, it should be detected. | Medium | S3.FC16 | - | Medium |
| Detective (COSO)  Detect (NIST CSF) | [S3.C4]  Monitor and investigate that all requests made with HTTP (e.g., via CloudTrail S3 data events with the lack of additionalEventData.CipherSuite) | Make an unencrypted AWS API call from one of your VPCs with VPC endpoint, it should be detected. | Low | S3.FC1  S3.FC5 | S3.T12 (Low)  S3.T34 (Low) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C18, depends on S3.C17]  For each VPC with an IAM entity allowed to use S3, secure them with the VPC ThreatModel (e.g. [modification of VPC endpoints, VPC endpoint policy, routing table, Security Groups](https://docs.aws.amazon.com/vpc/latest/userguide/vpce-gateway.html)) | Request how VPC ThreatModel for S3 is being applied. | High | S3.FC1  S3.FC5 | S3.T9 (Medium)  S3.T11 (Medium) | Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C25, depends on S3.C21,S3.C22]  Monitor VPC DNS query logs that only authorized S3 bucket and S3 access points are being queried in each VPC (e.g. using VPC DNS query logging), and protect it using Route53 ThreatModel | Make a DNS query to an unauthorized 1) S3 bucket and 2) S3 access points, it should be detected. | Low | S3.FC1  S3.FC5 | S3.T8 (Low)  S3.T9 (Low)  S3.T11 (Low) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C124]  Ensure all S3 VPC endpoints (Interface and Gateway) are covered by the VPC endpoints controls | Request the mechanism ensuring all S3 VPC endpoints (Interface and Gateway) are covered by the VPC endpoints controls, and its records | Low | S3.FC1 | S3.T45 (Very High) | Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C33, depends on S3.C31]  Monitor that only authorized AWS accounts to provide KMS keys are used for each AWS account (using CloudTrail S3 data events in *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Make a call to an unauthorized bucket, it should be detected | Low | S3.FC1  S3.FC15  S3.FC26  S3.FC5  S3.FC8 | S3.T1 (Low)  S3.T2 (Low)  S3.T4 (Low)  S3.T5 (Low)  S3.T7 (Low)  S3.T8 (Low)  S3.T9 (Low)  S3.T11 (Low)  S3.T16 (Low)  S3.T21 (Low)  S3.T27 (Very Low)  S3.T28 (Low)  S3.T30 (Low)  S3.T31 (Low) | Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C35]  Monitor ObjectACL changed (or tentatively changed) to public using CloudTrail S3 data events | Make a call to create a public ObjectACL, it should be detected. | Low | S3.FC1  S3.FC5 | S3.T6 (Low)  S3.T36 (Low) | Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C37]  Monitor and investigate anonymous requests to objects (e.g. using CloudTrail S3 data events with userIdentity.accountId=ANONYMOUS\_PRINCIPAL) | Make an anonymous call, it should be detected. | Low | S3.FC5 | S3.T36 (Low) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C41]  Parameterize the S3 bucket name or S3 access point in your code (no hardcoding) | Request the process on ensuring S3 bucket name or S3 access point are not hard-coded | Medium | S3.FC5 | S3.T1 (Low) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C42]  When connecting to S3 endpoints, use virtual-hosted model ("my-bucket-name.s3.amazonaws.com" or "my-bucket-name.my-s3-regional-endpoint.amazonaws.com") instead of path-style model ("s3.amazonaws.com/my-bucket-name" or "my-s3-regional-endpoint.amazonaws.com/my-bucket-name") (see [ref](https://aws.amazon.com/blogs/aws/amazon-s3-path-deprecation-plan-the-rest-of-the-story/)). All the latest SDKs make use of domain style by default. | Request the mechanism ensuring the usage of domain style instead of path style. | Very Low | S3.FC1 | S3.T35 (High) | Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C56]  Monitor changes on bucket ACL to ensure it stays private (e.g. using CloudTrail event PutBucketAcl and its fields requestParameters.x-amz-acl should be either "private" or not existing) | Make a call to have a bucket ACL other than private, it should be detected. | Medium | S3.FC19  S3.FC8 | S3.T4 (Low)  S3.T58 (Low) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C63, depends on S3.C61]  Use KMS ThreatModel to protect the KMS keys used for S3 (e.g. using encryptionContext on the policy of each KMS key) | Request the KMS ThreatModel and the evidence of its application to protect S3 | High | S3.FC1  S3.FC10  S3.FC5 | S3.T17 (Medium)  S3.T36 (Low)  S3.T37 (Low) | Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C68, depends on S3.C61]  Monitor that only authorized KMS key(s) are used on each bucket (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-sIDe-encryption-AWS-kms-key-ID*) | Make a request encrypted with an unauthorized KMS key, it should be detected | Low | S3.FC5 | S3.T11 (Very Low)  S3.T16 (Low)  S3.T30 (Very Low)  S3.T36 (Low) | Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C148, depends on S3.C145]  For buckets (or paths) requiring SSE-C, monitor that only authorized encryption is used on each bucket or path (using CloudTrail S3 data events in *requestParameter.bucketName* and *response.x-amz-server-side-encryption-customer-algorithm*) | Make a request to a bucket (or path) requiring SSE-C without the proper encryption, it should be detected | Low | S3.FC5 | S3.T11 (Very Low)  S3.T16 (Low)  S3.T30 (Very Low)  S3.T36 (Low) | Low |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C78, assured by S3.C79]  Reduce costs related to incomplete multipart upload by creating a lifecycle policy to remove them after an agreed length of time (e.g. 7 days) ([blog](https://aws.amazon.com/blogs/aws/s3-lifecycle-management-update-support-for-multipart-uploads-and-delete-markers/)) | Create an incomplete upload, and wait for the agreed time, it should be deleted automatically. | Low | S3.FC5 | S3.T40 (High) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C79]  Verify a lifecycle policy on incomplete multipart uploads is implemented on all buckets (e.g. using AWS Config rule: [S3\_LIFECYCLE\_POLICY\_CHECK](https://docs.aws.amazon.com/config/latest/developerguide/s3-lifecycle-policy-check.html)) | Create a bucket without a lifecycle policy to remove incomplete multipart upload, it should be detected | Medium | S3.FC5 | - | Low |
| Preventative (COSO)  Protect (NIST CSF) | [S3.C81]  Block all requests not using SigV4 (e.g. using an SCP and S3 policy on all buckets with deny on "StringNotEquals":{"s3:signatureversion": "AWS4-HMAC-SHA256"}) | Make a non-SigV4 AWS API call, it should be denied. | Low | S3.FC1 | S3.T35 (High) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C83]  Use SDK with SigV4 enabled ([ref](https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingAWSSDK.html#UsingAWSSDK-move-to-Sig4)) | Request the mechanism ensuring the use of SDK with SigV4 enabled | Low | S3.FC1 | S3.T35 (High) | Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C116]  Monitor abnormal behavior on replication CloudWatch metrics (i.e. *BytesPendingReplication* and *OperationsPendingReplication*) | Create an abnormal replication, it should be detected | Low | S3.FC15 | S3.T2 (Low) | Low |
| Directive (COSO)  Identify (NIST CSF) | [S3.C94]  Maintain a list of authorized notification receiver(s) (e.g. SNS topic, Lambda, etc.) for each bucket. You might use a simpler approach by using authorized account ID(s) to ensure all your receivers are in authorized AWS account(s). | Request the list of authorized notification receiver (e.g. SNS topic, Lambda, etc.) for each bucket, its review process, and its review records | Low | S3.FC20 | S3.T41 (Very Low) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C95]  Verify only authorized notification receiver(s) are configured for buckets. | Create an unauthorized receiver, it should be detected. | High | S3.FC20 | - | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C135, depends on S3.C94, assured by S3.C95]  Ensure only authorized notification receiver(s) (e.g. SNS topic, Lambda, etc.) for each bucket are configured | Request 1) the mechanism ensuring only authorized notification receiver(s) (e.g. SNS topic, Lambda, etc.) for each bucket are configured, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Medium | S3.FC20 | S3.T41 (High) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C103]  Protect and/or claim your domains and trademarks/copyrights (by creating your trademark buckets and using the [copyright infringement process](https://aws.amazon.com/terms/#notice-and-procedure-for-making-claims-of-copyright-infringement) from AWS) | Request the process by protecting and/or claiming your domains and trademarks/copyrights | Medium | S3.FC28 | S3.T23 (High) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C132, assured by S3.C133]  Ensure CloudWatch is enabled for all Object Lambda access points | Request the mechanism ensuring CloudWatch is enabled for all Object Lambda access points, and its records of execution | Low | S3.FC32 | S3.T46 (Low) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C133]  Verify CloudWatch is enabled for all Object Lambda access points | Create an Object Lambda access point without CloudWatch enabled, it should be detected. | Low | S3.FC32 | - | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C29]  Use an unguessable naming convention for the email addresses of your AWS accounts (e.g. add a + sign and a random string to redirect the email in the same mailbox) | Review naming convention for root account email and their implementation | Medium | S3.FC28 | S3.T19 (High) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C30]  Use an unguessable naming convention for your IAM users and IAM roles (e.g. add a random string) | Review naming convention for IAM users/role and their implementation | Medium | S3.FC28 | S3.T24 (High) | Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C155, assured by S3.C156]  Ensure all requests are blocked from unauthorized service roles (e.g. by denying all requests with the principal "arn:aws:iam::\*:\*/AWSServiceRoleFor\*" on S3 bucket policies) | Request 1) the mechanism ensuring only authorized service roles can access each bucket, 2) its records of execution for all new bucket, and 3) plan to move any older bucket | Low | S3.FC28 | S3.T57 (Very High) | Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C156]  Verify all requests are blocked from unauthorized service roles. | Remove the statement on a S3 bucket policy denying all unauthorized service roles, it should be detected. | Medium | S3.FC28 | - | Low |
| Directive (COSO)  Detect (NIST CSF) | [S3.C21]  Enable [VPC DNS query logging](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resolver-query-logs.html) in all VPC | Request the mechanism to enable VPC DNS query logging in all VPC | Medium | S3.FC1  S3.FC5 | S3.T8 (Very Low)  S3.T9 (Very Low)  S3.T11 (Very Low) | Very Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C40]  Scan your CNAME records (e.g. in Route53) and CloudFront origin for deleted buckets | Create a CNAME record and CloudFront origin with an invalid bucket, it should be detected. | High | S3.FC5 | S3.T1 (Very Low) | Very Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C43]  Monitor that all S3 connections are made with the virtual-hosted model (e.g. via CloudTrail S3 requestParameters.Host) | Make a path-style request to S3, it should be detected. | Medium | S3.FC1 | S3.T35 (Low) | Very Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C45]  Do not include sensitive data in bucket names, access point names, object names, object metadata and tags. | Request the process ensuring no sensitive data is included in bucket names, object names, object metadata and tags. | Low | S3.FC12  S3.FC20 | S3.T41 (Low)  S3.T42 (Medium) | Very Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C60]  Use a data discovery tool (e.g. Amazon Macie) to ensure the bucket names, object names, tags, and metadata do not contain sensitive data | Create a bucket name, object name, tags, or a metadata of an object with sensitive data, it should be detected. | Very High | S3.FC5 | S3.T11 (Very Low) | Very Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C80]  Block deprecated S3 actions using IAM ThreatModel and the S3 actions list. | Request the controls blocking deprecated S3 actions | Low | S3.FC1 | S3.T35 (Medium) | Very Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C82]  Monitor and investigate that all requests not using SigV4 (e.g. via CloudTrail S3 with the additionalEventData.SignatureVersion different from "SigV4") | Make a non-SigV4 AWS API call, it should be detected. | Low | S3.FC1 | S3.T35 (Low) | Very Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C85]  Monitor and investigate that all requests not using SigV4 (e.g. via CloudTrail S3 with the additionalEventData.SignatureVersion different from "SigV4") | Make 1) a presigned AWS API call and 2) a POST request, it should be detected. | Low | S3.FC5 | S3.T39 (Very Low) | Very Low |
| Directive (COSO)  Identify (NIST CSF) | [S3.C96]  Maintain a list of authorized S3 buckets to receive S3 inventory of each bucket | Request the list of authorized bucket(s) to receive S3 inventory of each bucket, its review process, and its review records | Low | S3.FC12 | S3.T42 (Very Low) | Very Low |
| Assurance (COSO)  Detect (NIST CSF) | [S3.C97]  Verify only authorized buckets are configured to receive inventory. | Create an unauthorized bucket to receive inventory, it should be detected. | High | S3.FC12 | - | Very Low |
| Directive (COSO)  Protect (NIST CSF) | [S3.C136, depends on S3.C96, assured by S3.C97]  Ensure only authorized S3 buckets are configured to receive S3 inventory for each bucket | Request 1) the mechanism ensuring only authorized S3 buckets are configured to receive S3 inventory for each bucket, 2) its records of execution for all new buckets, and 3) plan to move any older buckets | Medium | S3.FC12 | S3.T42 (Medium) | Very Low |
| Detective (COSO)  Detect (NIST CSF) | [S3.C157]  Monitor PutBucketLogging to detect bucket logging changes including deactivation and bucket change (i.e. using CloudTrail event *PutBucketLogging* and *requestParameter.BucketLoggingStatus* field to examine the lack of "LoggingEnabled" key, or an unauthorized bucket in requestParameters.BucketLoggingStatus.LoggingEnabled.TargetBucket) | Make a call to 1) disable bucket logging, or 2) change to an unauthorized bucket, it should be detected. | Low | S3.FC19 | S3.T59 (Medium) | Very Low |

## Appendix 2 - List of all Actions and their details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Id** | **Description** | **Feature Class ID** | **IAM Permission** | **Event** | **API** |
| S3.A1 | Aborts a multipart upload | S3.FC1 | s3:AbortMultipartUpload | AbortMultipartUpload | AbortMultipartUpload |
| S3.A2 | Grants permission to allow circumvention of governance-mode object retention settings (for DeleteObject, DeleteObjects and PutObjectRetention) | S3.FC17 | s3:BypassGovernanceRetention | - | - |
| S3.A3 | Completes a multipart upload by assembling previously uploaded parts | S3.FC1 | s3:PutObject | CompleteMultipartUpload | CompleteMultipartUpload |
| S3.A4 | Creates a copy of an object that is already stored in Amazon S3 | S3.FC1 | s3:GetObject  s3:PutObject | CopyObject | CopyObject |
| S3.A5 | Creates a new bucket | S3.FC5 | s3:CreateBucket | CreateBucket | CreateBucket |
| S3.A6 | Initiates a multipart upload and returns an upload ID | S3.FC1 | s3:GetObject  s3:PutObject | CreateMultipartUpload | CreateMultipartUpload |
| S3.A7 | Deletes the bucket. All objects (including all object versions and delete markers) in the bucket must be deleted before the bucket itself can be deleted | S3.FC5 | s3:DeleteBucket | DeleteBucket | DeleteBucket |
| S3.A8 | Deletes an analytics configuration for the bucket | S3.FC11 | s3:PutAnalyticsConfiguration | DeleteBucketAnalyticsConfiguration | DeleteBucketAnalyticsConfiguration |
| S3.A9 | Deletes the CORS configuration information set for the bucket | S3.FC22 | s3:PutBucketCORS | DeleteBucketCors | DeleteBucketCors |
| S3.A10 | Removes default encryption from the bucket | S3.FC23 | s3:PutEncryptionConfiguration | DeleteBucketEncryption | DeleteBucketEncryption |
| S3.A11 | Deletes an inventory configuration from the bucket | S3.FC12 | s3:PutInventoryConfiguration | DeleteBucketInventoryConfiguration | DeleteBucketInventoryConfiguration |
| S3.A12 | Deletes the lifecycle configuration from the bucket | S3.FC13 | s3:PutLifecycleConfiguration | DeleteBucketLifecycle | DeleteBucketLifecycle |
| S3.A13 | Deletes a metrics configuration for the Amazon CloudWatch request metrics (specified by the metrics configuration ID) from the bucket. Note that this doesn't include the daily storage metrics. | S3.FC14 | s3:PutMetricsConfiguration | DeleteBucketMetricsConfiguration | DeleteBucketMetricsConfiguration |
| S3.A14 | Deletes the policy on a specified bucket | S3.FC10 | s3:DeleteBucketPolicy | DeleteBucketPolicy | DeleteBucketPolicy |
| S3.A15 | Deletes the replication configuration from the bucket | S3.FC15 | s3:PutReplicationConfiguration | DeleteBucketReplication | DeleteBucketReplication |
| S3.A16 | Deletes the tags from the bucket. | S3.FC7 | s3:PutBucketTagging | DeleteBucketTagging | DeleteBucketTagging |
| S3.A17 | Removes the website configuration for a bucket | S3.FC16 | s3:DeleteBucketWebsite | DeleteBucketWebsite | DeleteBucketWebsite |
| S3.A18 | Deletes an object permanently (non-versioned bucket) or inserts a delete marker (versioned bucket) | S3.FC1 | s3:DeleteObject | DeleteObject | DeleteObject |
| S3.A19 | Permanently deletes an object or a delete marker from a bucket | S3.FC3 | s3:DeleteObjectVersion | DeleteObject | DeleteObject(VersionId=) |
| S3.A20 | Deletes multiple objects permanently (non-versioned bucket) or inserts delete markers (versioned bucket) | S3.FC1 | s3:DeleteObject | DeleteObjects | DeleteObjects |
| S3.A21 | Permanently deletes multiple objects or delete markers from a bucket | S3.FC3 | s3:DeleteObjectVersion | DeleteObjects | DeleteObjects(VersionId=) |
| S3.A22 | Removes the entire tag set from the specified object. | S3.FC2 | s3:DeleteObjectTagging | DeleteObjectTagging | DeleteObjectTagging |
| S3.A23 | Removes the entire tag set from the specified object version. | S3.FC4 | s3:DeleteObjectVersionTagging | DeleteObjectTagging | DeleteObjectTagging(VersionId=) |
| S3.A24 | Removes the PublicAccessBlock configuration for an Amazon S3 bucket. | S3.FC24 | s3:PutBucketPublicAccessBlock | DeletePublicAccessBlock | DeletePublicAccessBlock |
| S3.A25 | Returns the Transfer Acceleration state of a bucket, which is either "Enabled" or "Suspended". | S3.FC18 | s3:GetAccelerateConfiguration | GetBucketAccelerateConfiguration | GetBucketAccelerateConfiguration |
| S3.A26 | Returns the Access Control List (ACL) of a bucket | S3.FC8 | s3:GetBucketAcl | GetBucketAcl | GetBucketAcl |
| S3.A27 | Returns an analytics configuration from the bucket. | S3.FC11 | s3:GetAnalyticsConfiguration | GetBucketAnalyticsConfiguration | GetBucketAnalyticsConfiguration |
| S3.A28 | Returns the CORS configuration information set for the bucket. | S3.FC22 | s3:GetBucketCORS | GetBucketCors | GetBucketCors |
| S3.A29 | Returns the default encryption configuration for an Amazon S3 bucket. | S3.FC23 | s3:GetEncryptionConfiguration | GetBucketEncryption | GetBucketEncryption |
| S3.A30 | Returns an inventory configuration from the bucket. | S3.FC12 | s3:GetInventoryConfiguration | GetBucketInventoryConfiguration | GetBucketInventoryConfiguration |
| S3.A31 | (Deprecated) Returns the lifecycle configuration information set on the bucket. | S3.FC13 | s3:GetLifecycleConfiguration | GetBucketLifecycle | GetBucketLifecycle |
| S3.A32 | Returns the lifecycle configuration information set on the bucket. | S3.FC13 | s3:GetLifecycleConfiguration | GetBucketLifecycleConfiguration | GetBucketLifecycleConfiguration |
| S3.A33 | Returns a bucket's region. | S3.FC5 | s3:GetBucketLocation | GetBucketLocation | GetBucketLocation |
| S3.A34 | Returns the logging status of a bucket and the permissions users have to view and modify that status. | S3.FC19 | s3:GetBucketLogging | GetBucketLogging | GetBucketLogging |
| S3.A35 | Gets a metrics configuration from the bucket. | S3.FC14 | s3:GetMetricsConfiguration | GetBucketMetricsConfiguration | GetBucketMetricsConfiguration |
| S3.A36 | (Deprecated) Returns the notification configuration of a bucket. | S3.FC20 | s3:GetBucketNotification | GetBucketNotification | GetBucketNotification |
| S3.A37 | Returns the notification configuration of a bucket. | S3.FC20 | s3:GetBucketNotification | GetBucketNotificationConfiguration | GetBucketNotificationConfiguration |
| S3.A38 | Returns the policy of a specified bucket. | S3.FC10 | s3:GetBucketPolicy | GetBucketPolicy | GetBucketPolicy |
| S3.A39 | Retrieves the policy status for an Amazon S3 bucket, indicating whether the bucket is public. | S3.FC10 | s3:GetBucketPolicyStatus | GetBucketPolicyStatus | GetBucketPolicyStatus |
| S3.A40 | Returns the replication configuration of a bucket. | S3.FC15 | s3:GetReplicationConfiguration | GetBucketReplication | GetBucketReplication |
| S3.A41 | Returns the request payment configuration of a bucket. | S3.FC5 | s3:GetBucketRequestPayment | GetBucketRequestPayment | GetBucketRequestPayment |
| S3.A42 | Returns the tag set associated with the bucket. | S3.FC7 | s3:GetBucketTagging | GetBucketTagging | GetBucketTagging |
| S3.A43 | Returns the versioning state of a bucket. | S3.FC6 | s3:GetBucketVersioning | GetBucketVersioning | GetBucketVersioning |
| S3.A44 | Returns the website configuration for a bucket. | S3.FC16 | s3:GetBucketWebsite | GetBucketWebsite | GetBucketWebsite |
| S3.A45 | Retrieves an object from Amazon S3. | S3.FC1 | s3:GetObject | GetObject | GetObject |
| S3.A46 | Retrieves an object version from Amazon S3. | S3.FC3 | s3:GetObjectVersion | GetObject | GetObject(VersionId=) |
| S3.A47 | Returns ACL information about an object | S3.FC1 | s3:GetObjectAcl | GetObjectAcl | GetObjectAcl |
| S3.A48 | Returns ACL information about an object version, use the versionId subresource. | S3.FC9 | s3:GetObjectVersionAcl | GetObjectAcl | GetObjectAcl(VersionId=) |
| S3.A49 | Gets Object Lock legal hold for a specific object | S3.FC29 | s3:GetObjectLegalHold | GetObjectLegalHold | GetObjectLegalHold |
| S3.A50 | Gets the default S3 Object Lock configuration for a bucket. | S3.FC17 | s3:GetBucketObjectLockConfiguration | GetObjectLockConfiguration | GetObjectLockConfiguration |
| S3.A51 | Retrieves an object's retention settings. | S3.FC17 | s3:GetObjectRetention | GetObjectRetention | GetObjectRetention |
| S3.A52 | Returns the tag-set of an object. | S3.FC2 | s3:GetObjectTagging | GetObjectTagging | GetObjectTagging |
| S3.A53 | Returns the tag-set of a specific version of an object. | S3.FC4 | s3:GetObjectVersionTagging | GetObjectTagging | GetObjectTagging(VersionId=) |
| S3.A54 | Returns torrent files from an object. | S3.FC21 | s3:GetObjectTorrent | GetObjectTorrent | GetObjectTorrent |
| S3.A55 | (Deprecated) No documented usage of this action. | S3.FC21 | s3:GetObjectVersionTorrent | - | - |
| S3.A56 | Grants Amazon S3 the permission to replicate both unencrypted objects and objects encrypted with SSE-S3 or SSE-KMS | S3.FC15 | s3:GetObjectVersionForReplication | - | - |
| S3.A57 | Retrieves the PublicAccessBlock configuration for an Amazon S3 bucket. | S3.FC24 | s3:GetBucketPublicAccessBlock | GetPublicAccessBlock | GetPublicAccessBlock |
| S3.A58 | Determines if a bucket exists and you have permission to access it. | S3.FC1 | s3:HeadBucket | HeadBucket | HeadBucket |
| S3.A59 | Retrieves metadata from an object without returning the object itself. | S3.FC1 | s3:GetObject | HeadObject | HeadObject |
| S3.A60 | Retrieves metadata from an object version without returning the object itself. | S3.FC3 | s3:GetObjectVersion | HeadObject | HeadObject(VersionId=) |
| S3.A61 | Lists the analytics configurations for the bucket. | S3.FC11 | s3:GetAnalyticsConfiguration | ListBucketAnalyticsConfigurations | ListBucketAnalyticsConfigurations |
| S3.A62 | Returns a list of inventory configurations for the bucket. | S3.FC12 | s3:GetInventoryConfiguration | ListBucketInventoryConfigurations | ListBucketInventoryConfigurations |
| S3.A63 | Lists the metrics configurations for the bucket. | S3.FC14 | s3:GetMetricsConfiguration | ListBucketMetricsConfigurations | ListBucketMetricsConfigurations |
| S3.A64 | Returns a list of all buckets owned by the authenticated sender of the request. | S3.FC5 | s3:ListAllMyBuckets | ListBuckets | ListBuckets |
| S3.A65 | Lists in-progress multipart uploads. | S3.FC1 | s3:ListBucketMultipartUploads | ListMultipartUploads | ListMultipartUploads |
| S3.A66 | (Deprecated) Returns some or all (up to 1000) of the objects in a bucket. | S3.FC1 | s3:ListBucket | ListObjects | ListObjects |
| S3.A67 | Returns some or all (up to 1000) of the objects in a bucket. | S3.FC1 | s3:ListBucket | ListObjectsV2 | ListObjectsV2 |
| S3.A68 | Lists metadata about all of the versions of objects in a bucket. | S3.FC3 | s3:ListBucketVersions | ListObjectVersions | ListObjectVersions |
| S3.A69 | Lists the parts that have been uploaded for a specific multipart upload. | S3.FC1 | s3:ListMultipartUploadParts | ListParts | ListParts |
| S3.A70 | Allows Amazon S3 to change the ownership of a replicated object | S3.FC15 | s3:ObjectOwnerOverrideToBucketOwner | - | - |
| S3.A71 | Sets the Transfer Acceleration state of an existing bucket. | S3.FC18 | s3:PutAccelerateConfiguration | PutBucketAccelerateConfiguration | PutBucketAccelerateConfiguration |
| S3.A72 | Sets the permissions on an existing bucket using Access Control Lists (ACL). | S3.FC8 | s3:PutBucketAcl | PutBucketAcl | PutBucketAcl |
| S3.A73 | Adds an analytics configuration (identified by the analytics ID) to the bucket. | S3.FC11 | s3:PutAnalyticsConfiguration | PutBucketAnalyticsConfiguration | PutBucketAnalyticsConfiguration |
| S3.A74 | Sets the CORS configuration for your bucket. | S3.FC22 | s3:PutBucketCORS | PutBucketCors | PutBucketCors |
| S3.A75 | Sets the default encryption configuration for the bucket. | S3.FC23 | s3:PutEncryptionConfiguration | PutBucketEncryption | PutBucketEncryption |
| S3.A76 | Adds an inventory configuration (identified by the inventory ID) to the bucket | S3.FC12 | s3:PutInventoryConfiguration | PutBucketInventoryConfiguration | PutBucketInventoryConfiguration |
| S3.A77 | (Deprecated) Creates a new lifecycle configuration for the bucket or replaces an existing lifecycle configuration. | S3.FC13 | s3:PutLifecycleConfiguration | PutBucketLifecycle | PutBucketLifecycle |
| S3.A78 | Creates a new lifecycle configuration for the bucket or replaces an existing lifecycle configuration. | S3.FC13 | s3:PutLifecycleConfiguration | PutBucketLifecycleConfiguration | PutBucketLifecycleConfiguration |
| S3.A79 | Sets the logging parameters for a bucket. | S3.FC19 | s3:PutBucketLogging | PutBucketLogging | PutBucketLogging |
| S3.A80 | Sets or updates a metrics configuration for the CloudWatch request metrics (specified by the metrics configuration ID) from the bucket. | S3.FC14 | s3:PutMetricsConfiguration | PutBucketMetricsConfiguration | PutBucketMetricsConfiguration |
| S3.A81 | (Deprecated) Enables you to receive notifications when certain events happen in your bucket. | S3.FC20 | s3:PutBucketNotification | PutBucketNotification | PutBucketNotification |
| S3.A82 | Enables you to receive notifications when certain events happen in your bucket. | S3.FC20 | s3:PutBucketNotification | PutBucketNotificationConfiguration | PutBucketNotificationConfiguration |
| S3.A83 | Adds to or replaces a policy on a bucket. | S3.FC10 | s3:PutBucketPolicy | PutBucketPolicy | PutBucketPolicy |
| S3.A84 | Creates a new replication configuration (or replaces an existing one, if present). | S3.FC15 | s3:PutReplicationConfiguration | PutBucketReplication | PutBucketReplication |
| S3.A85 | Sets the request payment configuration of a bucket. | S3.FC5 | s3:PutBucketRequestPayment | PutBucketRequestPayment | PutBucketRequestPayment |
| S3.A86 | Adds a set of tags to an existing bucket. | S3.FC7 | s3:PutBucketTagging | PutBucketTagging | PutBucketTagging |
| S3.A87 | Sets the versioning state of an existing bucket. | S3.FC6 | s3:PutBucketVersioning | PutBucketVersioning | PutBucketVersioning |
| S3.A88 | Sets the configuration of the website that is specified in the website subresource. | S3.FC16 | s3:PutBucketWebsite | PutBucketWebsite | PutBucketWebsite |
| S3.A89 | Adds an object to a bucket. | S3.FC1 | s3:PutObject | PutObject,PostObject | PutObject |
| S3.A90 | Sets the Access Control List (ACL) permissions for an object. You must have WRITE\_ACP permission to set the ACL of an object. | S3.FC1 | s3:PutObjectAcl | PutObjectAcl | PutObjectAcl |
| S3.A91 | Sets the Access Control List (ACL) permissions for an object version. You must have WRITE\_ACP permission to set the ACL of an object version. | S3.FC9 | s3:PutObjectVersionAcl | PutObjectAcl | PutObjectAcl(VersionId=) |
| S3.A92 | Puts Object Lock legal hold on a specific object | S3.FC29 | s3:PutObjectLegalHold | PutObjectLegalHold | PutObjectLegalHold |
| S3.A93 | Allows placing a default S3 Object Lock configuration at bucket creation (AWS Support needs to be contacted for existing buckets). It automatically enables versioning, even without permission. | S3.FC17 | s3:PutBucketObjectLockConfiguration | PutObjectLockConfiguration | PutObjectLockConfiguration |
| S3.A94 | Puts object retention on a specific object | S3.FC17 | s3:PutObjectRetention | PutObjectRetention | PutObjectRetention |
| S3.A95 | Adds a set of tags to an existing object. | S3.FC2 | s3:PutObjectTagging | PutObjectTagging | PutObjectTagging |
| S3.A96 | Adds a set of tags to an existing object version | S3.FC4 | s3:PutObjectVersionTagging | PutObjectVersionTagging | PutObjectTagging(VersionId=) |
| S3.A97 | Creates or modifies the PublicAccessBlock configuration for an Amazon S3 bucket. | S3.FC24 | s3:PutBucketPublicAccessBlock | PutPublicAccessBlock | PutPublicAccessBlock |
| S3.A98 | Allows Amazon S3 to replicate delete markers to the destination bucket | S3.FC15 | s3:ReplicateDelete | - | - |
| S3.A99 | Allows Amazon S3 to replicate objects to the destination bucket, including tags | S3.FC15 | s3:ReplicateObject | - | - |
| S3.A100 | Allows Amazon S3 to replicate object tags to the destination bucket | S3.FC15 | s3:ReplicateTags | - | - |
| S3.A101 | Restores a temporary copy of an archived object. | S3.FC1 | s3:RestoreObject | RestoreObject | RestoreObject |
| S3.A102 | Filters the contents of an Amazon S3 object based on a simple structured query language (SQL) statement. | S3.FC1 | s3:GetObject | SelectObjectContent | SelectObjectContent |
| S3.A103 | Uploads a part in a multipart upload. | S3.FC1 | s3:PutObject | UploadPart | UploadPart |
| S3.A104 | Uploads a part by copying data from an existing object as a data source. | S3.FC1 | s3:PutObject  s3:GetObject | UploadPartCopy | UploadPartCopy |
| S3.A105 | Creates a new access point. | S3.FC26 | s3:CreateAccessPoint | CreateAccessPoint | CreateAccessPoint |
| S3.A106 | Creates a new Amazon S3 Batch Operations job. | S3.FC27 | s3:CreateJob | JobCreated | CreateJob |
| S3.A107 | Deletes the specified access point. | S3.FC26 | s3:DeleteAccessPoint | DeleteAccessPoint | DeleteAccessPoint |
| S3.A108 | Deletes the policy on a specified access point | S3.FC26 | s3:DeleteAccessPointPolicy | DeleteAccessPointPolicy | DeleteAccessPointPolicy |
| S3.A109 | Removes the PublicAccessBlock configuration for an AWS account. | S3.FC25 | s3:PutAccountPublicAccessBlock | DeletePublicAccessBlock | DeletePublicAccessBlock |
| S3.A110 | Retrieves the configuration parameters and status for a Batch Operations job. | S3.FC27 | s3:DescribeJob | DescribeJob | DescribeJob |
| S3.A111 | Retrieves access point metadata | S3.FC26 | s3:GetAccessPoint | GetAccessPoint | GetAccessPoint |
| S3.A112 | Returns the policy of a specified access point. | S3.FC26 | s3:GetAccessPointPolicy | GetAccessPointPolicy | GetAccessPointPolicy |
| S3.A113 | Retrieves the policy status for a specific access point's policy | S3.FC26 | s3:GetAccessPointPolicyStatus | GetAccessPointPolicyStatus | GetAccessPointPolicyStatus |
| S3.A114 | Retrieves the PublicAccessBlock configuration for an AWS account | S3.FC25 | s3:GetAccountPublicAccessBlock | GetPublicAccessBlock | GetPublicAccessBlock |
| S3.A115 | Returns a list of the access points currently associated with the specified bucket. | S3.FC26 | s3:ListAccessPoints | ListAccessPoints | ListAccessPoints |
| S3.A116 | Lists current jobs and jobs that have ended recently. | S3.FC27 | s3:ListJobs | ListJobs | ListJobs |
| S3.A117 | Adds to or replaces a data policy on an access point. | S3.FC26 | s3:PutAccessPointPolicy | PutAccessPointPolicy | PutAccessPointPolicy |
| S3.A118 | Creates or modifies the PublicAccessBlock configuration for an AWS account. | S3.FC25 | s3:PutAccountPublicAccessBlock | PutPublicAccessBlock | PutPublicAccessBlock |
| S3.A119 | Updates an existing job's priority. | S3.FC27 | s3:UpdateJobPriority | - | UpdateJobPriority |
| S3.A120 | Updates the status for the specified job. | S3.FC27 | s3:UpdateJobStatus | JobStatusChanged | UpdateJobStatus |
| S3.A121 | Removes OwnershipControls for an Amazon S3 bucket. | S3.FC30 | s3:PutBucketOwnershipControls | DeleteBucketOwnershipControls | DeleteBucketOwnershipControls |
| S3.A122 | Retrieves OwnershipControls for an Amazon S3 bucket. | S3.FC30 | s3:GetBucketOwnershipControls | GetBucketOwnershipControls | GetBucketOwnershipControls |
| S3.A123 | Creates or modifies OwnershipControls for an Amazon S3 bucket | S3.FC30 | s3:PutBucketOwnershipControls | PutBucketOwnershipControls | PutBucketOwnershipControls |
| S3.A124 | Deletes the S3 Intelligent-Tiering configuration from the specified bucket | S3.FC13 | s3:DeleteIntelligentTieringConfiguration | DeleteBucketIntelligentTieringConfiguration | DeleteBucketIntelligentTieringConfiguration |
| S3.A125 | Gets the S3 Intelligent-Tiering configuration from the specified bucket | S3.FC13 | s3:GetIntelligentTieringConfiguration | GetBucketIntelligentTieringConfiguration | GetBucketIntelligentTieringConfiguration |
| S3.A126 | Lists the S3 Intelligent-Tiering configuration from the specified bucket | S3.FC13 | s3:ListIntelligentTieringConfigurations | ListBucketIntelligentTieringConfigurations | ListBucketIntelligentTieringConfigurations |
| S3.A127 | Puts a S3 Intelligent-Tiering configuration to the specified bucket | S3.FC13 | s3:PutIntelligentTieringConfiguration | PutBucketIntelligentTieringConfiguration | PutBucketIntelligentTieringConfiguration |
| S3.A128 | Deletes the Amazon S3 Storage Lens configuration | S3.FC31 | s3:DeleteStorageLensConfiguration | DeleteStorageLensConfiguration | DeleteStorageLensConfiguration |
| S3.A129 | Deletes the Amazon S3 Storage Lens configuration tags | S3.FC31 | s3:DeleteStorageLensConfigurationTagging | DeleteStorageLensConfigurationTagging | DeleteStorageLensConfigurationTagging |
| S3.A130 | Gets the Amazon S3 Storage Lens configuration | S3.FC31 | s3:GetStorageLensConfiguration | GetStorageLensConfiguration | GetStorageLensConfiguration |
| S3.A131 | Gets the tags of Amazon S3 Storage Lens configuration | S3.FC31 | s3:GetStorageLensConfigurationTagging | GetStorageLensConfiguratioTagging | GetStorageLensConfiguratioTagging |
| S3.A132 | Gets a list of Amazon S3 Storage Lens configurations | S3.FC31 | s3:ListStorageLensConfigurations | ListStorageLensConfigurations | ListStorageLensConfigurations |
| S3.A133 | Puts an Amazon S3 Storage Lens configuration | S3.FC31 | s3:PutStorageLensConfiguration | PutStorageLensConfiguration | PutStorageLensConfiguration |
| S3.A134 | Puts or replaces tags on an existing Amazon S3 Storage Lens configuration | S3.FC31 | s3:PutStorageLensConfigurationTagging | PutStorageLensConfigurationTagging | PutStorageLensConfigurationTagging |
| S3.A135 | Creates an Object Lambda access point | S3.FC32 | s3:CreateAccessPointForObjectLambda | CreateAccessPointForObjectLambda | CreateAccessPointForObjectLambda |
| S3.A136 | Deletes the specified Object Lambda access point | S3.FC32 | s3:DeleteAccessPointForObjectLambda | DeleteAccessPointForObjectLambda | DeleteAccessPointForObjectLambda |
| S3.A137 | Removes the resource policy for an Object Lambda access point | S3.FC32 | s3:DeleteAccessPointPolicyForObjectLambda | DeleteAccessPointPolicyForObjectLambda | DeleteAccessPointPolicyForObjectLambda |
| S3.A138 | Returns configuration for an Object Lambda access point | S3.FC32 | s3:GetAccessPointConfigurationForObjectLambda | GetAccessPointConfigurationForObjectLambda | GetAccessPointConfigurationForObjectLambda |
| S3.A139 | Returns configuration information about the specified Object Lambda access point | S3.FC32 | s3:GetAccessPointForObjectLambda | GetAccessPointForObjectLambda | GetAccessPointForObjectLambda |
| S3.A140 | Returns the resource policy for an Object Lambda access point | S3.FC32 | s3:GetAccessPointPolicyForObjectLambda | GetAccessPointPolicyForObjectLambda | GetAccessPointPolicyForObjectLambda |
| S3.A141 | Returns the status of the resource policy associated with an Object Lambda access point | S3.FC32 | s3:GetAccessPointPolicyStatusForObjectLambda | GetAccessPointPolicyStatusForObjectLambda | GetAccessPointPolicyStatusForObjectLambda |
| S3.A142 | Returns a list of the access points associated with the Object Lambda access point | S3.FC32 | s3:ListAccessPointsForObjectLambda | ListAccessPointsForObjectLambda | ListAccessPointsForObjectLambda |
| S3.A143 | Replaces configuration for an Object Lambda access point | S3.FC32 | s3:PutAccessPointConfigurationForObjectLambda | PutAccessPointConfigurationForObjectLambda | PutAccessPointConfigurationForObjectLambda |
| S3.A144 | Creates or replaces resource policy for an Object Lambda access point | S3.FC32 | s3:PutAccessPointPolicyForObjectLambda | PutAccessPointPolicyForObjectLambda | PutAccessPointPolicyForObjectLambda |
| S3.A145 | Grants permission to abort a multipart upload | S3.FC32 | s3-object-lambda:AbortMultipartUpload | - | - |
| S3.A146 | Grants permission to remove the null version of an object and insert a delete marker, which becomes the current version of the object | S3.FC32 | s3-object-lambda:DeleteObject | - | - |
| S3.A147 | Grants permission to use the tagging subresource to remove the entire tag set from the specified object | S3.FC32 | s3-object-lambda:DeleteObjectTagging | - | - |
| S3.A148 | Grants permission to retrieve objects from Amazon S3 | S3.FC32 | s3-object-lambda:GetObject | - | - |
| S3.A149 | Grants permission to return the Access Control List (ACL) of an object | S3.FC32 | s3-object-lambda:GetObjectAcl | - | - |
| S3.A150 | Grants permission to get an object's current legal hold status | S3.FC32 | s3-object-lambda:GetObjectLegalHold | - | - |
| S3.A151 | Grants permission to retrieve the retention settings for an object | S3.FC32 | s3-object-lambda:GetObjectRetention | - | - |
| S3.A152 | Grants permission to return the tag set of an object | S3.FC32 | s3-object-lambda:GetObjectTagging | - | - |
| S3.A153 | Grants permission to retrieve a specific version of an object | S3.FC32 | s3-object-lambda:GetObjectVersion | - | - |
| S3.A154 | Grants permission to list some or all of the objects in an Amazon S3 bucket (up to 1000) | S3.FC32 | s3-object-lambda:ListBucket | - | - |
| S3.A155 | Grants permission to list the parts that have been uploaded for a specific multipart upload | S3.FC32 | s3-object-lambda:ListMultipartUploadParts | - | - |
| S3.A156 | Grants permission to add an object to a bucket | S3.FC32 | s3-object-lambda:PutObject | - | - |
| S3.A157 | Grants permission to set the Access Control List (ACL) permissions for new or existing objects in an S3 bucket. | S3.FC32 | s3-object-lambda:PutObjectAcl | - | - |
| S3.A158 | Grants permission to apply a legal hold configuration to the specified object | S3.FC32 | s3-object-lambda:PutObjectLegalHold | - | - |
| S3.A159 | Grants permission to place an object retention configuration on an object | S3.FC32 | s3-object-lambda:PutObjectRetention | - | - |
| S3.A160 | Grants permission to set the supplied tag-set to an object that already exists in a bucket | S3.FC32 | s3-object-lambda:PutObjectTagging | - | - |
| S3.A161 | Grants permission to restore an archived copy of an object back into Amazon S3 | S3.FC32 | s3-object-lambda:RestoreObject | - | - |
| S3.A162 | Passes transformed objects to a GetObject operation when using Object Lambda access points | S3.FC32 | s3-object-lambda:WriteGetObjectResponse | WriteGetObjectResponse | WriteGetObjectResponse |
| S3.A163 | Grants permission to remove a specific version of an object | S3.FC32 | s3-object-lambda:DeleteObjectVersion | - | - |
| S3.A164 | Grants permission to remove the entire tag set for a specific version of the object | S3.FC32 | s3-object-lambda:DeleteObjectVersionTagging | - | - |
| S3.A165 | Grants permission to return the Access Control List (ACL) of a specific object version | S3.FC32 | s3-object-lambda:GetObjectVersionAcl | - | - |
| S3.A166 | Grants permission to return the tag set for a specific version of the object | S3.FC32 | s3-object-lambda:GetObjectVersionTagging | - | - |
| S3.A167 | Grants permission to list in-progress multipart uploads | S3.FC32 | s3-object-lambda:ListBucketMultipartUploads | - | - |
| S3.A168 | Grants permission to list metadata about all the versions of objects in an Amazon S3 bucket | S3.FC32 | s3-object-lambda:ListBucketVersions | - | - |
| S3.A169 | Grants permission to use the ACL subresource to set the Access Control List (ACL) permissions for an object that already exists in a bucket | S3.FC32 | s3-object-lambda:PutObjectVersionAcl | - | - |
| S3.A170 | Grants permission to set the supplied tag-set for a specific version of an object | S3.FC32 | s3-object-lambda:PutObjectVersionTagging | - | - |
| S3.A171 | Returns configuration information about the specified Multi-Region Access Point. | S3.FC33 | s3:GetMultiRegionAccessPoint | GetMultiRegionAccessPoint | GetMultiRegionAccessPoint |
| S3.A172 | Indicates whether the specified Multi-Region Access Point has an access control policy that allows public access. | S3.FC33 | s3:GetMultiRegionAccessPointPolicyStatus | GetMultiRegionAccessPointPolicyStatus | GetMultiRegionAccessPointPolicyStatus |
| S3.A173 | Creates a Multi-Region Access Point and associates it with the specified buckets. | S3.FC33 | s3:CreateMultiRegionAccessPoint | CreateMultiRegionAccessPoint | CreateMultiRegionAccessPoint |
| S3.A174 | Retrieves the status of an asynchronous request to manage a Multi-Region Access Point. | S3.FC33 | s3:DescribeMultiRegionAccessPointOperation | DescribeMultiRegionAccessPointOperation | DescribeMultiRegionAccessPointOperation |
| S3.A175 | Deletes a Multi-Region Access Point. This action does not delete the buckets associated with the Multi-Region Access Point, only the Multi-Region Access Point itself. | S3.FC33 | s3:DeleteMultiRegionAccessPoint | DeleteMultiRegionAccessPoint | DeleteMultiRegionAccessPoint |
| S3.A176 | Returns a list of the Multi-Region Access Points currently associated with the specified AWS account. | S3.FC33 | s3:ListMultiRegionAccessPoints | ListMultiRegionAccessPoints | ListMultiRegionAccessPoints |
| S3.A177 | Returns the access control policy of the specified Multi-Region Access Point. | S3.FC33 | s3:GetMultiRegionAccessPointPolicy | GetMultiRegionAccessPointPolicy | GetMultiRegionAccessPointPolicy |
| S3.A178 | Associates an access control policy with the specified Multi-Region Access Point. | S3.FC33 | s3:PutMultiRegionAccessPointPolicy | PutMultiRegionAccessPointPolicy | PutMultiRegionAccessPointPolicy |
| S3.A179 | Remove tags from an existing Amazon S3 Batch Operations job | S3.FC27 | s3:DeleteJobTagging | DeleteJobTagging | DeleteJobTagging |
| S3.A180 | Return the tag set of an existing Amazon S3 Batch Operations job | S3.FC27 | s3:GetJobTagging | GetJobTagging | GetJobTagging |
| S3.A181 | Get an Amazon S3 Storage Lens dashboard | S3.FC31 | s3:GetStorageLensDashboard | GetStorageLensDashboardDataInternal | GetStorageLensDashboard |
| S3.A182 | Replace tags on an existing Amazon S3 Batch Operations job | S3.FC27 | s3:PutJobTagging | PutJobTagging | PutJobTagging |
| S3.A183 | Associate Public Access Block configurations with a specified access point, while creating a access point | S3.FC26 | s3:PutAccessPointPublicAccessBlock | PutAccessPointPublicAccessBlock | PutAccessPointPublicAccessBlock |
| S3.A184 | Initiate the replication process by setting replication status of an object to pending | S3.FC27 | s3:InitiateReplication | - | - |
| S3.A185 | Retrieves all the metadata from an object without returning the object itself. This action is useful if you're interested only in an object's metadata. | S3.FC1 | s3:GetObjectAttributes  s3:GetObject | GetObjectAttributes | GetObjectAttributes |
| S3.A186 | Retrieves all the metadata from a versioned object without returning the object itself. This action is useful if you're interested only in an object's metadata. | S3.FC9 | s3:GetObjectVersionAttributes  s3:GetObjectVersion | GetObjectAttributes | GetObjectAttributes(VersionId=) |

## Appendix 3 - List of the Service availability in AWS Regions

|  |  |  |  |
| --- | --- | --- | --- |
| **Region Name** | **Region Id** | **Amazon Simple Storage Service** (s3) | **AWS S3 Control** (s3-control) |
| Africa (Cape Town) | af-south-1 | **Yes** | **No** |
| Asia Pacific (Hong Kong) | ap-east-1 | **Yes** | **No** |
| Asia Pacific (Tokyo) | ap-northeast-1 | **Yes** | **Yes** |
| Asia Pacific (Seoul) | ap-northeast-2 | **Yes** | **Yes** |
| Asia Pacific (Osaka) | ap-northeast-3 | **Yes** | **Yes** |
| Asia Pacific (Mumbai) | ap-south-1 | **Yes** | **Yes** |
| Asia Pacific (Singapore) | ap-southeast-1 | **Yes** | **Yes** |
| Asia Pacific (Sydney) | ap-southeast-2 | **Yes** | **Yes** |
| Asia Pacific (Jakarta) | ap-southeast-3 | **Yes** | **No** |
| Canada (Central) | ca-central-1 | **Yes** | **Yes** |
| China (Beijing) | cn-north-1 | **Yes** | **Yes** |
| China (Ningxia) | cn-northwest-1 | **Yes** | **Yes** |
| Europe (Frankfurt) | eu-central-1 | **Yes** | **Yes** |
| Europe (Stockholm) | eu-north-1 | **Yes** | **Yes** |
| Europe (Milan) | eu-south-1 | **Yes** | **No** |
| Europe (Ireland) | eu-west-1 | **Yes** | **Yes** |
| Europe (London) | eu-west-2 | **Yes** | **Yes** |
| Europe (Paris) | eu-west-3 | **Yes** | **Yes** |
| Middle East (UAE) | me-central-1 | **Yes** | **No** |
| Middle East (Bahrain) | me-south-1 | **Yes** | **No** |
| South America (São Paulo) | sa-east-1 | **Yes** | **Yes** |
| US East (N. Virginia) | us-east-1 | **Yes** | **Yes** |
| US East (Ohio) | us-east-2 | **Yes** | **Yes** |
| AWS GovCloud (US-East) | us-gov-east-1 | **Yes** | **Yes** |
| AWS GovCloud (US-West) | us-gov-west-1 | **Yes** | **Yes** |
| US ISO East | us-iso-east-1 | **Yes** | **No** |
| US ISO WEST | us-iso-west-1 | **Yes** | **No** |
| US ISOB East (Ohio) | us-isob-east-1 | **Yes** | **No** |
| US West (N. California) | us-west-1 | **Yes** | **Yes** |
| US West (Oregon) | us-west-2 | **Yes** | **Yes** |