

Package ‘paws.storage’

February 8, 2023

Title 'Amazon Web Services' Storage Services

Version 0.2.0

Description Interface to 'Amazon Web Services' storage services, including 'Simple Storage Service' ('S3') and more
<<https://aws.amazon.com/>>.

License Apache License (>= 2.0)

URL <https://github.com/paws-r/paws>

BugReports <https://github.com/paws-r/paws/issues>

Imports paws.common (>= 0.5.4)

Suggests testthat

Encoding UTF-8

RoxygenNote 7.2.2

Collate 'backup_service.R' 'backup_interfaces.R' 'backup_operations.R'
'backupstorage_service.R' 'backupstorage_interfaces.R'
'backupstorage_operations.R' 'dlm_service.R' 'dlm_interfaces.R'
'dlm_operations.R' 'ebs_service.R' 'ebs_interfaces.R'
'ebs_operations.R' 'efs_service.R' 'efs_interfaces.R'
'efs_operations.R' 'finspacedata_service.R'
'finspacedata_interfaces.R' 'finspacedata_operations.R'
'fsx_service.R' 'fsx_interfaces.R' 'fsx_operations.R'
'glacier_service.R' 'glacier_interfaces.R'
'glacier_operations.R' 'recyclebin_service.R'
'recyclebin_interfaces.R' 'recyclebin_operations.R'
's3_service.R' 's3_operations.R' 's3_custom.R'
's3_interfaces.R' 's3control_service.R'
's3control_interfaces.R' 's3control_operations.R'
's3outposts_service.R' 's3outposts_interfaces.R'
's3outposts_operations.R' 'storagegateway_service.R'
'storagegateway_interfaces.R' 'storagegateway_operations.R'

NeedsCompilation no

Author David Kretch [aut],
 Adam Banker [aut],
 Dyfan Jones [cre],
 Amazon.com, Inc. [cph]

Maintainer Dyfan Jones <dyfan.r.jones@gmail.com>

Repository CRAN

Date/Publication 2023-02-08 13:10:09 UTC

R topics documented:

backup	2
backupstorage	5
d1m	7
ebs	9
efs	11
finspacedata	13
fsx	15
glacier	17
recyclebin	20
s3	22
s3control	26
s3outposts	29
storagegateway	30
Index	35

backup	<i>AWS Backup</i>
--------	-------------------

Description

Backup

Backup is a unified backup service designed to protect Amazon Web Services services and their associated data. Backup simplifies the creation, migration, restoration, and deletion of backups, while also providing reporting and auditing.

Usage

```
backup(config = list())
```

Arguments

config	Optional configuration of credentials, endpoint, and/or region. <ul style="list-style-type: none"> • access_key_id: AWS access key ID • secret_access_key: AWS secret access key • session_token: AWS temporary session token • profile: The name of a profile to use. If not given, then the default profile is used. • anonymous: Set anonymous credentials. • endpoint: The complete URL to use for the constructed client. • region: The AWS Region used in instantiating the client. • close_connection: Immediately close all HTTP connections. • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds. • s3_force_path_style: Set this to <code>true</code> to force the request to use path-style addressing, i.e., <code>http://s3.amazonaws.com/BUCKET/KEY</code>.
--------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- backup(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)
```

Operations

[create_backup_plan](#)

Creates a backup plan using a backup plan name and backup rules

<code>create_backup_selection</code>	Creates a JSON document that specifies a set of resources to assign to a backup plan
<code>create_backup_vault</code>	Creates a logical container where backups are stored
<code>create_framework</code>	Creates a framework with one or more controls
<code>create_report_plan</code>	Creates a report plan
<code>delete_backup_plan</code>	Deletes a backup plan
<code>delete_backup_selection</code>	Deletes the resource selection associated with a backup plan that is specified by the plan ID
<code>delete_backup_vault</code>	Deletes the backup vault identified by its name
<code>delete_backup_vault_access_policy</code>	Deletes the policy document that manages permissions on a backup vault
<code>delete_backup_vault_lock_configuration</code>	Deletes Backup Vault Lock from a backup vault specified by a backup vault name
<code>delete_backup_vault_notifications</code>	Deletes event notifications for the specified backup vault
<code>delete_framework</code>	Deletes the framework specified by a framework name
<code>delete_recovery_point</code>	Deletes the recovery point specified by a recovery point ID
<code>delete_report_plan</code>	Deletes the report plan specified by a report plan name
<code>describe_backup_job</code>	Returns backup job details for the specified BackupJobId
<code>describe_backup_vault</code>	Returns metadata about a backup vault specified by its name
<code>describe_copy_job</code>	Returns metadata associated with creating a copy of a resource
<code>describe_framework</code>	Returns the framework details for the specified FrameworkName
<code>describe_global_settings</code>	Describes whether the Amazon Web Services account is opted in to cross-account backup
<code>describe_protected_resource</code>	Returns information about a saved resource, including the last time it was backed up
<code>describe_recovery_point</code>	Returns metadata associated with a recovery point, including ID, status, encryption, and storage container
<code>describe_region_settings</code>	Returns the current service opt-in settings for the Region
<code>describe_report_job</code>	Returns the details associated with creating a report as specified by its ReportJobId
<code>describe_report_plan</code>	Returns a list of all report plans for an Amazon Web Services account and Amazon Resource Name
<code>describe_restore_job</code>	Returns metadata associated with a restore job that is specified by a job ID
<code>disassociate_recovery_point</code>	Deletes the specified continuous backup recovery point from Backup and releases the resource
<code>export_backup_plan_template</code>	Returns the backup plan that is specified by the plan ID as a backup template
<code>get_backup_plan</code>	Returns BackupPlan details for the specified BackupPlanId
<code>get_backup_plan_from_json</code>	Returns a valid JSON document specifying a backup plan or an error
<code>get_backup_plan_from_template</code>	Returns the template specified by its templateId as a backup plan
<code>get_backup_selection</code>	Returns selection metadata and a document in JSON format that specifies a list of resources to back up
<code>get_backup_vault_access_policy</code>	Returns the access policy document that is associated with the named backup vault
<code>get_backup_vault_notifications</code>	Returns event notifications for the specified backup vault
<code>get_recovery_point_restore_metadata</code>	Returns a set of metadata key-value pairs that were used to create the backup
<code>get_supported_resource_types</code>	Returns the Amazon Web Services resource types supported by Backup
<code>list_backup_jobs</code>	Returns a list of existing backup jobs for an authenticated account for the last 30 days
<code>list_backup_plans</code>	Returns a list of all active backup plans for an authenticated account
<code>list_backup_plan_templates</code>	Returns metadata of your saved backup plan templates, including the template ID, name, and description
<code>list_backup_plan_versions</code>	Returns version metadata of your backup plans, including Amazon Resource Name, version number, and creation date
<code>list_backup_selections</code>	Returns an array containing metadata of the resources associated with the target backup plan
<code>list_backup_vaults</code>	Returns a list of recovery point storage containers along with information about the vault
<code>list_copy_jobs</code>	Returns metadata about your copy jobs
<code>list_frameworks</code>	Returns a list of all frameworks for an Amazon Web Services account and Amazon Resource Name
<code>list_protected_resources</code>	Returns an array of resources successfully backed up by Backup, including the time they were backed up
<code>list_recovery_points_by_backup_vault</code>	Returns detailed information about the recovery points stored in a backup vault
<code>list_recovery_points_by_resource</code>	Returns detailed information about all the recovery points of the type specified by the resource type
<code>list_report_jobs</code>	Returns details about your report jobs
<code>list_report_plans</code>	Returns a list of your report plans
<code>list_restore_jobs</code>	Returns a list of jobs that Backup initiated to restore a saved resource, including details about the job

list_tags	Returns a list of key-value pairs assigned to a target recovery point, backup plan, or backup vault
put_backup_vault_access_policy	Sets a resource-based policy that is used to manage access permissions on the target backup vault
put_backup_vault_lock_configuration	Applies Backup Vault Lock to a backup vault, preventing attempts to delete any recovery points
put_backup_vault_notifications	Turns on notifications on a backup vault for the specified topic and events
start_backup_job	Starts an on-demand backup job for the specified resource
start_copy_job	Starts a job to create a one-time copy of the specified resource
start_report_job	Starts an on-demand report job for the specified report plan
start_restore_job	Recovers the saved resource identified by an Amazon Resource Name (ARN)
stop_backup_job	Attempts to cancel a job to create a one-time backup of a resource
tag_resource	Assigns a set of key-value pairs to a recovery point, backup plan, or backup vault
untag_resource	Removes a set of key-value pairs from a recovery point, backup plan, or backup vault
update_backup_plan	Updates an existing backup plan identified by its backupPlanId with the input document
update_framework	Updates an existing framework identified by its FrameworkName with the input document
update_global_settings	Updates whether the Amazon Web Services account is opted in to cross-account backup
update_recovery_point_lifecycle	Sets the transition lifecycle of a recovery point
update_region_settings	Updates the current service opt-in settings for the Region
update_report_plan	Updates an existing report plan identified by its ReportPlanName with the input document

Examples

```
## Not run:
svc <- backup()
svc$create_backup_plan(
  Foo = 123
)

## End(Not run)
```

backupstorage	<i>AWS Backup Storage</i>
---------------	---------------------------

Description

The frontend service for Cryo Storage.

Usage

```
backupstorage(config = list())
```

Arguments

`config` Optional configuration of credentials, endpoint, and/or region.

- **access_key_id**: AWS access key ID
- **secret_access_key**: AWS secret access key

- **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e., `http://s3.amazonaws.com/BUCKET/KEY`.

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- backupstorage(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)
```

Operations

delete_object	Delete Object from the incremental base Backup
get_chunk	Gets the specified object's chunk
get_object_metadata	Get metadata associated with an Object
list_chunks	List chunks in a given Object
list_objects	List all Objects in a given Backup
notify_object_complete	Complete upload
put_chunk	Upload chunk

`put_object`
`start_object`

Upload object that can store object metadata String and data blob in single API call using inline chunking.
 Start upload containing one or many chunks

Examples

```
## Not run:
svc <- backupstorage()
svc$delete_object(
  Foo = 123
)

## End(Not run)
```

d1m

Amazon Data Lifecycle Manager

Description

With Amazon Data Lifecycle Manager, you can manage the lifecycle of your Amazon Web Services resources. You create lifecycle policies, which are used to automate operations on the specified resources.

Amazon DLM supports Amazon EBS volumes and snapshots. For information about using Amazon DLM with Amazon EBS, see [Automating the Amazon EBS Snapshot Lifecycle](#) in the *Amazon EC2 User Guide*.

Usage

```
d1m(config = list())
```

Arguments

`config` Optional configuration of credentials, endpoint, and/or region.

- **access_key_id**: AWS access key ID
- **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e., `http://s3.amazonaws.com/BUCKET/KEY`.

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- dlm(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)
```

Operations

create_lifecycle_policy	Creates a policy to manage the lifecycle of the specified Amazon Web Services resources
delete_lifecycle_policy	Deletes the specified lifecycle policy and halts the automated operations that the policy specified
get_lifecycle_policies	Gets summary information about all or the specified data lifecycle policies
get_lifecycle_policy	Gets detailed information about the specified lifecycle policy
list_tags_for_resource	Lists the tags for the specified resource
tag_resource	Adds the specified tags to the specified resource
untag_resource	Removes the specified tags from the specified resource
update_lifecycle_policy	Updates the specified lifecycle policy

Examples

```
## Not run:
svc <- dlm()
svc$create_lifecycle_policy(
  Foo = 123
)

## End(Not run)
```


Description

You can use the Amazon Elastic Block Store (Amazon EBS) direct APIs to create Amazon EBS snapshots, write data directly to your snapshots, read data on your snapshots, and identify the differences or changes between two snapshots. If you're an independent software vendor (ISV) who offers backup services for Amazon EBS, the EBS direct APIs make it more efficient and cost-effective to track incremental changes on your Amazon EBS volumes through snapshots. This can be done without having to create new volumes from snapshots, and then use Amazon Elastic Compute Cloud (Amazon EC2) instances to compare the differences.

You can create incremental snapshots directly from data on-premises into volumes and the cloud to use for quick disaster recovery. With the ability to write and read snapshots, you can write your on-premises data to an snapshot during a disaster. Then after recovery, you can restore it back to Amazon Web Services or on-premises from the snapshot. You no longer need to build and maintain complex mechanisms to copy data to and from Amazon EBS.

This API reference provides detailed information about the actions, data types, parameters, and errors of the EBS direct APIs. For more information about the elements that make up the EBS direct APIs, and examples of how to use them effectively, see [Accessing the Contents of an Amazon EBS Snapshot](#) in the *Amazon Elastic Compute Cloud User Guide*. For more information about the supported Amazon Web Services Regions, endpoints, and service quotas for the EBS direct APIs, see [Amazon Elastic Block Store Endpoints and Quotas](#) in the *Amazon Web Services General Reference*.

Usage

```
ebs(config = list())
```

Arguments

config	Optional configuration of credentials, endpoint, and/or region. <ul style="list-style-type: none">• access_key_id: AWS access key ID• secret_access_key: AWS secret access key• session_token: AWS temporary session token• profile: The name of a profile to use. If not given, then the default profile is used.• anonymous: Set anonymous credentials.• endpoint: The complete URL to use for the constructed client.• region: The AWS Region used in instantiating the client.• close_connection: Immediately close all HTTP connections.• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e., <code>http://s3.amazonaws.com/BUCKET/KEY</code>.
--------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- ebs(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)
```

Operations

complete_snapshot	Seals and completes the snapshot after all of the required blocks of data have been written to it
get_snapshot_block	Returns the data in a block in an Amazon Elastic Block Store snapshot
list_changed_blocks	Returns information about the blocks that are different between two Amazon Elastic Block Store snapshots
list_snapshot_blocks	Returns information about the blocks in an Amazon Elastic Block Store snapshot
put_snapshot_block	Writes a block of data to a snapshot
start_snapshot	Creates a new Amazon EBS snapshot

Examples

```
## Not run:
svc <- ebs()
svc$complete_snapshot(
  Foo = 123
)

## End(Not run)
```

Description

Amazon Elastic File System (Amazon EFS) provides simple, scalable file storage for use with Amazon EC2 Linux and Mac instances in the Amazon Web Services Cloud. With Amazon EFS, storage capacity is elastic, growing and shrinking automatically as you add and remove files, so that your applications have the storage they need, when they need it. For more information, see the [Amazon Elastic File System API Reference](#) and the [Amazon Elastic File System User Guide](#).

Usage

```
efs(config = list())
```

Arguments

config	Optional configuration of credentials, endpoint, and/or region. <ul style="list-style-type: none"> • access_key_id: AWS access key ID • secret_access_key: AWS secret access key • session_token: AWS temporary session token • profile: The name of a profile to use. If not given, then the default profile is used. • anonymous: Set anonymous credentials. • endpoint: The complete URL to use for the constructed client. • region: The AWS Region used in instantiating the client. • close_connection: Immediately close all HTTP connections. • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds. • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e., <code>http://s3.amazonaws.com/BUCKET/KEY</code>.
--------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- efs(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
```

```

        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical"
)
)

```

Operations

create_access_point	Creates an EFS access point
create_file_system	Creates a new, empty file system
create_mount_target	Creates a mount target for a file system
create_replication_configuration	Creates a replication configuration that replicates an existing EFS file system to a new file system
create_tags	DEPRECATED - CreateTags is deprecated and not maintained
delete_access_point	Deletes the specified access point
delete_file_system	Deletes a file system, permanently severing access to its contents
delete_file_system_policy	Deletes the FileSystemPolicy for the specified file system
delete_mount_target	Deletes the specified mount target
delete_replication_configuration	Deletes an existing replication configuration
delete_tags	DEPRECATED - DeleteTags is deprecated and not maintained
describe_access_points	Returns the description of a specific Amazon EFS access point if the AccessPointID is specified
describe_account_preferences	Returns the account preferences settings for the Amazon Web Services account associated with the EFS file system
describe_backup_policy	Returns the backup policy for the specified EFS file system
describe_file_system_policy	Returns the FileSystemPolicy for the specified EFS file system
describe_file_systems	Returns the description of a specific Amazon EFS file system if either the file system ID or the file system name is specified
describe_lifecycle_configuration	Returns the current LifecycleConfiguration object for the specified Amazon EFS file system
describe_mount_targets	Returns the descriptions of all the current mount targets, or a specific mount target, for the specified file system
describe_mount_target_security_groups	Returns the security groups currently in effect for a mount target
describe_replication_configurations	Retrieves the replication configuration for a specific file system
describe_tags	DEPRECATED - The DescribeTags action is deprecated and not maintained
list_tags_for_resource	Lists all tags for a top-level EFS resource
modify_mount_target_security_groups	Modifies the set of security groups in effect for a mount target
put_account_preferences	Use this operation to set the account preference in the current Amazon Web Services account
put_backup_policy	Updates the file system's backup policy
put_file_system_policy	Applies an Amazon EFS FileSystemPolicy to an Amazon EFS file system
put_lifecycle_configuration	Use this action to manage EFS lifecycle management and intelligent tiering
tag_resource	Creates a tag for an EFS resource
untag_resource	Removes tags from an EFS resource
update_file_system	Updates the throughput mode or the amount of provisioned throughput of an existing file system

Examples

```
## Not run:
svc <- efs()
# This operation creates a new, encrypted file system with automatic
# backups enabled, and the default generalpurpose performance mode.
svc$create_file_system(
  Backup = TRUE,
  CreationToken = "tokenstring",
  Encrypted = TRUE,
  PerformanceMode = "generalPurpose",
  Tags = list(
    list(
      Key = "Name",
      Value = "MyFileSystem"
    )
  )
)
## End(Not run)
```

finspacedata

*FinSpace Public API***Description**

The FinSpace APIs let you take actions inside the FinSpace.

Usage

```
finspacedata(config = list())
```

Arguments

config	Optional configuration of credentials, endpoint, and/or region. <ul style="list-style-type: none"> • access_key_id: AWS access key ID • secret_access_key: AWS secret access key • session_token: AWS temporary session token • profile: The name of a profile to use. If not given, then the default profile is used. • anonymous: Set anonymous credentials. • endpoint: The complete URL to use for the constructed client. • region: The AWS Region used in instantiating the client. • close_connection: Immediately close all HTTP connections. • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds. • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e., <code>http://s3.amazonaws.com/BUCKET/KEY</code>.
--------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- finspacedata(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)
```

Operations

associate_user_to_permission_group	Adds a user account to a permission group to grant permissions for actions a user can perform in FinSpace
create_changeset	Creates a new Changeset in a FinSpace Dataset
create_dataset	Creates a new FinSpace Dataset
create_data_view	Creates a Dataview for a Dataset
create_permission_group	Creates a group of permissions for various actions that a user can perform in FinSpace
create_user	Creates a new user in FinSpace
delete_dataset	Deletes a FinSpace Dataset
delete_permission_group	Deletes a permission group
disable_user	Denies access to the FinSpace web application and API for the specified user
disassociate_user_from_permission_group	Removes a user account from a permission group
enable_user	Allows the specified user to access the FinSpace web application and API
get_changeset	Get information about a Changeset
get_dataset	Returns information about a Dataset
get_data_view	Gets information about a Dataview
get_external_data_view_access_details	Returns the credentials to access the external Dataview from an S3 location
get_permission_group	Retrieves the details of a specific permission group
get_programmatic_access_credentials	Request programmatic credentials to use with FinSpace SDK
get_user	Retrieves details for a specific user
get_working_location	A temporary Amazon S3 location, where you can copy your files from a source location
list_changesets	Lists the FinSpace Changesets for a Dataset

list_datasets	Lists all of the active Datasets that a user has access to
list_data_views	Lists all available Dataviews for a Dataset
list_permission_groups	Lists all available permission groups in FinSpace
list_permission_groups_by_user	Lists all the permission groups that are associated with a specific user account
list_users	Lists all available user accounts in FinSpace
list_users_by_permission_group	Lists details of all the users in a specific permission group
reset_user_password	Resets the password for a specified user ID and generates a temporary one
update_changeset	Updates a FinSpace Changeset
update_dataset	Updates a FinSpace Dataset
update_permission_group	Modifies the details of a permission group
update_user	Modifies the details of the specified user account

Examples

```
## Not run:
svc <- finspacedata()
svc$associate_user_to_permission_group(
  Foo = 123
)

## End(Not run)
```

 fsx

Amazon FSx

Description

Amazon FSx is a fully managed service that makes it easy for storage and application administrators to launch and use shared file storage.

Usage

```
fsx(config = list())
```

Arguments

`config` Optional configuration of credentials, endpoint, and/or region.

- **access_key_id**: AWS access key ID
- **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.

- **region:** The AWS Region used in instantiating the client.
- **close_connection:** Immediately close all HTTP connections.
- **timeout:** The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style:** Set this to `true` to force the request to use path-style addressing, i.e., `http://s3.amazonaws.com/BUCKET/KEY`.

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- fsx(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)
```

Operations

associate_file_system_aliases	Use this action to associate one or more Domain Name Server (DNS) aliases with an Amazon FSx file system.
cancel_data_repository_task	Cancel an existing Amazon FSx for Lustre data repository task if that task is in either the <code>Running</code> or <code>Starting</code> state.
copy_backup	Copy an existing backup within the same Amazon Web Services account to another Amazon Web Services account.
create_backup	Create a backup of an existing Amazon FSx for Windows File Server file system, Amazon FSx for Lustre file system, or Amazon FSx for OpenZFS file system.
create_data_repository_association	Create an Amazon FSx for Lustre data repository association (DRA).
create_data_repository_task	Create an Amazon FSx for Lustre data repository task.
create_file_system	Create a new, empty Amazon FSx file system.
create_file_system_from_backup	Create a new Amazon FSx for Lustre, Amazon FSx for Windows File Server, or Amazon FSx for OpenZFS file system from a backup.
create_snapshot	Create a snapshot of an existing Amazon FSx for OpenZFS volume.
create_storage_virtual_machine	Create a storage virtual machine (SVM) for an Amazon FSx for ONTAP file system.
create_volume	Create an FSx for ONTAP or Amazon FSx for OpenZFS storage volume.
create_volume_from_backup	Create a new Amazon FSx for NetApp ONTAP volume from an existing Amazon FSx for ONTAP volume.

<code>delete_backup</code>	Deletes an Amazon FSx backup
<code>delete_data_repository_association</code>	Deletes a data repository association on an Amazon FSx for Lustre file system
<code>delete_file_system</code>	Deletes a file system
<code>delete_snapshot</code>	Deletes an Amazon FSx for OpenZFS snapshot
<code>delete_storage_virtual_machine</code>	Deletes an existing Amazon FSx for ONTAP storage virtual machine (SVM)
<code>delete_volume</code>	Deletes an Amazon FSx for NetApp ONTAP or Amazon FSx for OpenZFS volume
<code>describe_backups</code>	Returns the description of a specific Amazon FSx backup, if a BackupIds value is provided
<code>describe_data_repository_associations</code>	Returns the description of specific Amazon FSx for Lustre data repository associations
<code>describe_data_repository_tasks</code>	Returns the description of specific Amazon FSx for Lustre data repository tasks, if one or more DataRepositoryTaskIds are provided
<code>describe_file_system_aliases</code>	Returns the DNS aliases that are associated with the specified Amazon FSx for Windows File System (FSx for Windows)
<code>describe_file_systems</code>	Returns the description of specific Amazon FSx file systems, if a FileSystemIds value is provided
<code>describe_snapshots</code>	Returns the description of specific Amazon FSx for OpenZFS snapshots, if a SnapshotIds value is provided
<code>describe_storage_virtual_machines</code>	Describes one or more Amazon FSx for NetApp ONTAP storage virtual machines (SVMs)
<code>describe_volumes</code>	Describes one or more Amazon FSx for NetApp ONTAP or Amazon FSx for OpenZFS volumes
<code>disassociate_file_system_aliases</code>	Use this action to disassociate, or remove, one or more Domain Name Service (DNS) aliases from the specified Amazon FSx for Windows File System (FSx for Windows)
<code>list_tags_for_resource</code>	Lists tags for Amazon FSx resources
<code>release_file_system_nfs_v3_locks</code>	Releases the file system lock from an Amazon FSx for OpenZFS file system
<code>restore_volume_from_snapshot</code>	Returns an Amazon FSx for OpenZFS volume to the state saved by the specified snapshot
<code>tag_resource</code>	Tags an Amazon FSx resource
<code>untag_resource</code>	This action removes a tag from an Amazon FSx resource
<code>update_data_repository_association</code>	Updates the configuration of an existing data repository association on an Amazon FSx for Lustre file system
<code>update_file_system</code>	Use this operation to update the configuration of an existing Amazon FSx file system
<code>update_snapshot</code>	Updates the name of an Amazon FSx for OpenZFS snapshot
<code>update_storage_virtual_machine</code>	Updates an Amazon FSx for ONTAP storage virtual machine (SVM)
<code>update_volume</code>	Updates the configuration of an Amazon FSx for NetApp ONTAP or Amazon FSx for OpenZFS volume

Examples

```
## Not run:
svc <- fsx()
# This operation copies an Amazon FSx backup.
svc$copy_backup(
  SourceBackupId = "backup-03e3c82e0183b7b6b",
  SourceRegion = "us-east-2"
)

## End(Not run)
```

Description

Amazon S3 Glacier (Glacier) is a storage solution for "cold data."

Glacier is an extremely low-cost storage service that provides secure, durable, and easy-to-use storage for data backup and archival. With Glacier, customers can store their data cost effectively for months, years, or decades. Glacier also enables customers to offload the administrative burdens of operating and scaling storage to AWS, so they don't have to worry about capacity planning, hardware provisioning, data replication, hardware failure and recovery, or time-consuming hardware migrations.

Glacier is a great storage choice when low storage cost is paramount and your data is rarely retrieved. If your application requires fast or frequent access to your data, consider using Amazon S3. For more information, see [Amazon Simple Storage Service \(Amazon S3\)](#).

You can store any kind of data in any format. There is no maximum limit on the total amount of data you can store in Glacier.

If you are a first-time user of Glacier, we recommend that you begin by reading the following sections in the *Amazon S3 Glacier Developer Guide*:

- [What is Amazon S3 Glacier](#) - This section of the Developer Guide describes the underlying data model, the operations it supports, and the AWS SDKs that you can use to interact with the service.
- [Getting Started with Amazon S3 Glacier](#) - The Getting Started section walks you through the process of creating a vault, uploading archives, creating jobs to download archives, retrieving the job output, and deleting archives.

Usage

```
glacier(config = list())
```

Arguments

config	Optional configuration of credentials, endpoint, and/or region. <ul style="list-style-type: none"> • access_key_id: AWS access key ID • secret_access_key: AWS secret access key • session_token: AWS temporary session token • profile: The name of a profile to use. If not given, then the default profile is used. • anonymous: Set anonymous credentials. • endpoint: The complete URL to use for the constructed client. • region: The AWS Region used in instantiating the client. • close_connection: Immediately close all HTTP connections. • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds. • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e., <code>http://s3.amazonaws.com/BUCKET/KEY</code>.
--------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- glacier(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)
```

Operations

[abort_multipart_upload](#)
[abort_vault_lock](#)
[add_tags_to_vault](#)
[complete_multipart_upload](#)
[complete_vault_lock](#)
[create_vault](#)
[delete_archive](#)
[delete_vault](#)
[delete_vault_access_policy](#)
[delete_vault_notifications](#)
[describe_job](#)
[describe_vault](#)
[get_data_retrieval_policy](#)
[get_job_output](#)
[get_vault_access_policy](#)
[get_vault_lock](#)
[get_vault_notifications](#)
[initiate_job](#)
[initiate_multipart_upload](#)
[initiate_vault_lock](#)

This operation aborts a multipart upload identified by the upload ID
This operation aborts the vault locking process if the vault lock is not in the Locked state
This operation adds the specified tags to a vault
You call this operation to inform Amazon S3 Glacier (Glacier) that all the archive parts have been uploaded
This operation completes the vault locking process by transitioning the vault lock from the Incomplete state to the Locked state
This operation creates a new vault with the specified name
This operation deletes an archive from a vault
This operation deletes a vault
This operation deletes the access policy associated with the specified vault
This operation deletes the notification configuration set for a vault
This operation returns information about a job you previously initiated, including the job ID, the job name, and the job status
This operation returns information about a vault, including the vault's Amazon Resource Name (ARN), the vault's name, and the vault's status
This operation returns the current data retrieval policy for the account and region specified in the request
This operation downloads the output of the job you initiated using `InitiateJob`
This operation retrieves the access-policy subresource set on the vault; for more information, see [Access Policies](#)
This operation retrieves the following attributes from the lock-policy subresource set on the vault: `LockPolicyName`, `LockPolicyType`, `LockPolicyVersion`, and `LockPolicyStatus`
This operation retrieves the notification-configuration subresource of the specified vault
This operation initiates a job of the specified type, which can be a select, an archival retrieval, or a multipart upload
This operation initiates a multipart upload
This operation initiates the vault locking process by doing the following:

list_jobs	This operation lists jobs for a vault, including jobs that are in-progress and jobs that have re
list_multipart_uploads	This operation lists in-progress multipart uploads for the specified vault
list_parts	This operation lists the parts of an archive that have been uploaded in a specific multipart up
list_provisioned_capacity	This operation lists the provisioned capacity units for the specified AWS account
list_tags_for_vault	This operation lists all the tags attached to a vault
list_vaults	This operation lists all vaults owned by the calling user's account
purchase_provisioned_capacity	This operation purchases a provisioned capacity unit for an AWS account
remove_tags_from_vault	This operation removes one or more tags from the set of tags attached to a vault
set_data_retrieval_policy	This operation sets and then enacts a data retrieval policy in the region specified in the PUT
set_vault_access_policy	This operation configures an access policy for a vault and will overwrite an existing policy
set_vault_notifications	This operation configures notifications that will be sent when specific events happen to a va
upload_archive	This operation adds an archive to a vault
upload_multipart_part	This operation uploads a part of an archive

Examples

```
## Not run:
svc <- glacier()
# The example deletes an in-progress multipart upload to a vault named
# my-vault:
svc$abort_multipart_upload(
  accountId = "-",
  uploadId = "19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-OssZtLq...",
  vaultName = "my-vault"
)

## End(Not run)
```

recyclebin

Amazon Recycle Bin

Description

This is the *Recycle Bin API Reference*. This documentation provides descriptions and syntax for each of the actions and data types in Recycle Bin.

Recycle Bin is a resource recovery feature that enables you to restore accidentally deleted snapshots and EBS-backed AMIs. When using Recycle Bin, if your resources are deleted, they are retained in the Recycle Bin for a time period that you specify.

You can restore a resource from the Recycle Bin at any time before its retention period expires. After you restore a resource from the Recycle Bin, the resource is removed from the Recycle Bin, and you can then use it in the same way you use any other resource of that type in your account. If the retention period expires and the resource is not restored, the resource is permanently deleted from the Recycle Bin and is no longer available for recovery. For more information about Recycle Bin, see [Recycle Bin](#) in the *Amazon Elastic Compute Cloud User Guide*.

Usage

```
recyclebin(config = list())
```

Arguments

`config` Optional configuration of credentials, endpoint, and/or region.

- **access_key_id**: AWS access key ID
- **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e., `http://s3.amazonaws.com/BUCKET/KEY`.

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- recyclebin(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)
```

Operations

<code>create_rule</code>	Creates a Recycle Bin retention rule
<code>delete_rule</code>	Deletes a Recycle Bin retention rule
<code>get_rule</code>	Gets information about a Recycle Bin retention rule
<code>list_rules</code>	Lists the Recycle Bin retention rules in the Region
<code>list_tags_for_resource</code>	Lists the tags assigned to a retention rule
<code>tag_resource</code>	Assigns tags to the specified retention rule
<code>untag_resource</code>	Unassigns a tag from a retention rule
<code>update_rule</code>	Updates an existing Recycle Bin retention rule

Examples

```
## Not run:
svc <- recyclebin()
svc$create_rule(
  Foo = 123
)

## End(Not run)
```

s3

Amazon Simple Storage Service

Description

Amazon Simple Storage Service

Usage

```
s3(config = list())
```

Arguments

`config` Optional configuration of credentials, endpoint, and/or region.

- **access_key_id**: AWS access key ID
- **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.

- **close_connection:** Immediately close all HTTP connections.
- **timeout:** The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style:** Set this to `true` to force the request to use path-style addressing, i.e., `http://s3.amazonaws.com/BUCKET/KEY`.

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- s3(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)
```

Operations

[abort_multipart_upload](#)

[complete_multipart_upload](#)

[copy_object](#)

[create_bucket](#)

[create_multipart_upload](#)

[delete_bucket](#)

[delete_bucket_analytics_configuration](#)

[delete_bucket_cors](#)

[delete_bucket_encryption](#)

[delete_bucket_intelligent_tiering_configuration](#)

[delete_bucket_inventory_configuration](#)

[delete_bucket_lifecycle](#)

[delete_bucket_metrics_configuration](#)

This action aborts a multipart upload

Completes a multipart upload by assembling previously uploaded parts

Creates a copy of an object that is already stored in Amazon S3

Creates a new S3 bucket

This action initiates a multipart upload and returns an upload ID

Deletes the S3 bucket

Deletes an analytics configuration for the bucket (specified by the analytics

Deletes the cors configuration information set for the bucket

This implementation of the DELETE action removes default encryption from

Deletes the S3 Intelligent-Tiering configuration from the specified bucket

Deletes an inventory configuration (identified by the inventory ID) from the

Deletes the lifecycle configuration from the specified bucket

Deletes a metrics configuration for the Amazon CloudWatch request metrics

delete_bucket_ownership_controls	Removes OwnershipControls for an Amazon S3 bucket
delete_bucket_policy	This implementation of the DELETE action uses the policy subresource to c
delete_bucket_replication	Deletes the replication configuration from the bucket
delete_bucket_tagging	Deletes the tags from the bucket
delete_bucket_website	This action removes the website configuration for a bucket
delete_object	Removes the null version (if there is one) of an object and inserts a delete m
delete_objects	This action enables you to delete multiple objects from a bucket using a sing
delete_object_tagging	Removes the entire tag set from the specified object
delete_public_access_block	Removes the PublicAccessBlock configuration for an Amazon S3 bucket
download_file	Download a file from S3 and store it at a specified file location
generate_presigned_url	@title Generate a presigned url given a client, its method, and arguments
get_bucket_accelerate_configuration	This implementation of the GET action uses the accelerate subresource to r
get_bucket_acl	This implementation of the GET action uses the acl subresource to return th
get_bucket_analytics_configuration	This implementation of the GET action returns an analytics configuration (i
get_bucket_cors	Returns the Cross-Origin Resource Sharing (CORS) configuration informat
get_bucket_encryption	Returns the default encryption configuration for an Amazon S3 bucket
get_bucket_intelligent_tiering_configuration	Gets the S3 Intelligent-Tiering configuration from the specified bucket
get_bucket_inventory_configuration	Returns an inventory configuration (identified by the inventory configuratio
get_bucket_lifecycle	For an updated version of this API, see GetBucketLifecycleConfiguration
get_bucket_lifecycle_configuration	Bucket lifecycle configuration now supports specifying a lifecycle rule usin
get_bucket_location	Returns the Region the bucket resides in
get_bucket_logging	Returns the logging status of a bucket and the permissions users have to vie
get_bucket_metrics_configuration	Gets a metrics configuration (specified by the metrics configuration ID) fro
get_bucket_notification	No longer used, see GetBucketNotificationConfiguration
get_bucket_notification_configuration	Returns the notification configuration of a bucket
get_bucket_ownership_controls	Retrieves OwnershipControls for an Amazon S3 bucket
get_bucket_policy	Returns the policy of a specified bucket
get_bucket_policy_status	Retrieves the policy status for an Amazon S3 bucket, indicating whether the
get_bucket_replication	Returns the replication configuration of a bucket
get_bucket_request_payment	Returns the request payment configuration of a bucket
get_bucket_tagging	Returns the tag set associated with the bucket
get_bucket_versioning	Returns the versioning state of a bucket
get_bucket_website	Returns the website configuration for a bucket
get_object	Retrieves objects from Amazon S3
get_object_acl	Returns the access control list (ACL) of an object
get_object_attributes	Retrieves all the metadata from an object without returning the object itself
get_object_legal_hold	Gets an object's current legal hold status
get_object_lock_configuration	Gets the Object Lock configuration for a bucket
get_object_retention	Retrieves an object's retention settings
get_object_tagging	Returns the tag-set of an object
get_object_torrent	Returns torrent files from a bucket
get_public_access_block	Retrieves the PublicAccessBlock configuration for an Amazon S3 bucket
head_bucket	This action is useful to determine if a bucket exists and you have permission
head_object	The HEAD action retrieves metadata from an object without returning the o
list_bucket_analytics_configurations	Lists the analytics configurations for the bucket
list_bucket_intelligent_tiering_configurations	Lists the S3 Intelligent-Tiering configuration from the specified bucket
list_bucket_inventory_configurations	Returns a list of inventory configurations for the bucket
list_bucket_metrics_configurations	Lists the metrics configurations for the bucket

list_buckets	Returns a list of all buckets owned by the authenticated sender of the request
list_multipart_uploads	This action lists in-progress multipart uploads
list_objects	Returns some or all (up to 1,000) of the objects in a bucket
list_objects_v2	Returns some or all (up to 1,000) of the objects in a bucket with each request
list_object_versions	Returns metadata about all versions of the objects in a bucket
list_parts	Lists the parts that have been uploaded for a specific multipart upload
put_bucket_accelerate_configuration	Sets the accelerate configuration of an existing bucket
put_bucket_acl	Sets the permissions on an existing bucket using access control lists (ACL)
put_bucket_analytics_configuration	Sets an analytics configuration for the bucket (specified by the analytics configuration ID)
put_bucket_cors	Sets the cors configuration for your bucket
put_bucket_encryption	This action uses the encryption subresource to configure default encryption
put_bucket_intelligent_tiering_configuration	Puts a S3 Intelligent-Tiering configuration to the specified bucket
put_bucket_inventory_configuration	This implementation of the PUT action adds an inventory configuration (identified by the configuration ID)
put_bucket_lifecycle	For an updated version of this API, see PutBucketLifecycleConfiguration
put_bucket_lifecycle_configuration	Creates a new lifecycle configuration for the bucket or replaces an existing one
put_bucket_logging	Set the logging parameters for a bucket and to specify permissions for who can log
put_bucket_metrics_configuration	Sets a metrics configuration (specified by the metrics configuration ID) for the bucket
put_bucket_notification	No longer used, see the PutBucketNotificationConfiguration operation
put_bucket_notification_configuration	Enables notifications of specified events for a bucket
put_bucket_ownership_controls	Creates or modifies OwnershipControls for an Amazon S3 bucket
put_bucket_policy	Applies an Amazon S3 bucket policy to an Amazon S3 bucket
put_bucket_replication	Creates a replication configuration or replaces an existing one
put_bucket_request_payment	Sets the request payment configuration for a bucket
put_bucket_tagging	Sets the tags for a bucket
put_bucket_versioning	Sets the versioning state of an existing bucket
put_bucket_website	Sets the configuration of the website that is specified in the website subresource
put_object	Adds an object to a bucket
put_object_acl	Uses the acl subresource to set the access control list (ACL) permissions for the object
put_object_legal_hold	Applies a legal hold configuration to the specified object
put_object_lock_configuration	Places an Object Lock configuration on the specified bucket
put_object_retention	Places an Object Retention configuration on an object
put_object_tagging	Sets the supplied tag-set to an object that already exists in a bucket
put_public_access_block	Creates or modifies the PublicAccessBlock configuration for an Amazon S3 bucket
restore_object	Restores an archived copy of an object back into Amazon S3
select_object_content	This action filters the contents of an Amazon S3 object based on a simple string
upload_part	Uploads a part in a multipart upload
upload_part_copy	Uploads a part by copying data from an existing object as data source
write_get_object_response	Passes transformed objects to a GetObject operation when using Object Lambda

Examples

```
## Not run:
svc <- s3()
# The following example aborts a multipart upload.
svc$abort_multipart_upload(
  Bucket = "examplebucket",
  Key = "bigobject",
```

```

    UploadId = "xadc0B_7YPBOJuoFiQ9cz4P3Pe6FIZw04f7wN93uHsNBEw97p15eNwzExg0LA..."
)

## End(Not run)

```

s3control

AWS S3 Control

Description

Amazon Web Services S3 Control provides access to Amazon S3 control plane actions.

Usage

```
s3control(config = list())
```

Arguments

config	<p>Optional configuration of credentials, endpoint, and/or region.</p> <ul style="list-style-type: none"> • access_key_id: AWS access key ID • secret_access_key: AWS secret access key • session_token: AWS temporary session token • profile: The name of a profile to use. If not given, then the default profile is used. • anonymous: Set anonymous credentials. • endpoint: The complete URL to use for the constructed client. • region: The AWS Region used in instantiating the client. • close_connection: Immediately close all HTTP connections. • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds. • s3_force_path_style: Set this to <code>true</code> to force the request to use path-style addressing, i.e., <code>http://s3.amazonaws.com/BUCKET/KEY</code>.
--------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```

svc <- s3control(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)

```

Operations

create_access_point	Creates an access point and associates it with the specified bucket
create_access_point_for_object_lambda	Creates an Object Lambda Access Point
create_bucket	This action creates an Amazon S3 on Outposts bucket
create_job	You can use S3 Batch Operations to perform large-scale batch actions on
create_multi_region_access_point	Creates a Multi-Region Access Point and associates it with the specified
delete_access_point	Deletes the specified access point
delete_access_point_for_object_lambda	Deletes the specified Object Lambda Access Point
delete_access_point_policy	Deletes the access point policy for the specified access point
delete_access_point_policy_for_object_lambda	Removes the resource policy for an Object Lambda Access Point
delete_bucket	This action deletes an Amazon S3 on Outposts bucket
delete_bucket_lifecycle_configuration	This action deletes an Amazon S3 on Outposts bucket's lifecycle config
delete_bucket_policy	This action deletes an Amazon S3 on Outposts bucket policy
delete_bucket_tagging	This action deletes an Amazon S3 on Outposts bucket's tags
delete_job_tagging	Removes the entire tag set from the specified S3 Batch Operations job
delete_multi_region_access_point	Deletes a Multi-Region Access Point
delete_public_access_block	Removes the PublicAccessBlock configuration for an Amazon Web Ser
delete_storage_lens_configuration	Deletes the Amazon S3 Storage Lens configuration
delete_storage_lens_configuration_tagging	Deletes the Amazon S3 Storage Lens configuration tags
describe_job	Retrieves the configuration parameters and status for a Batch Operation
describe_multi_region_access_point_operation	Retrieves the status of an asynchronous request to manage a Multi-Regi
get_access_point	Returns configuration information about the specified access point
get_access_point_configuration_for_object_lambda	Returns configuration for an Object Lambda Access Point
get_access_point_for_object_lambda	Returns configuration information about the specified Object Lambda A
get_access_point_policy	Returns the access point policy associated with the specified access poi
get_access_point_policy_for_object_lambda	Returns the resource policy for an Object Lambda Access Point
get_access_point_policy_status	Indicates whether the specified access point currently has a policy that a

<code>get_access_point_policy_status_for_object_lambda</code>	Returns the status of the resource policy associated with an Object Lambda
<code>get_bucket</code>	Gets an Amazon S3 on Outposts bucket
<code>get_bucket_lifecycle_configuration</code>	This action gets an Amazon S3 on Outposts bucket's lifecycle configuration
<code>get_bucket_policy</code>	This action gets a bucket policy for an Amazon S3 on Outposts bucket
<code>get_bucket_tagging</code>	This action gets an Amazon S3 on Outposts bucket's tags
<code>get_job_tagging</code>	Returns the tags on an S3 Batch Operations job
<code>get_multi_region_access_point</code>	Returns configuration information about the specified Multi-Region Access Point
<code>get_multi_region_access_point_policy</code>	Returns the access control policy of the specified Multi-Region Access Point
<code>get_multi_region_access_point_policy_status</code>	Indicates whether the specified Multi-Region Access Point has an access control policy
<code>get_public_access_block</code>	Retrieves the PublicAccessBlock configuration for an Amazon Web Services resource
<code>get_storage_lens_configuration</code>	Gets the Amazon S3 Storage Lens configuration
<code>get_storage_lens_configuration_tagging</code>	Gets the tags of Amazon S3 Storage Lens configuration
<code>list_access_points</code>	Returns a list of the access points currently associated with the specified Multi-Region Access Point
<code>list_access_points_for_object_lambda</code>	Returns some or all (up to 1,000) access points associated with the Object Lambda
<code>list_jobs</code>	Lists current S3 Batch Operations jobs and jobs that have ended within the specified time period
<code>list_multi_region_access_points</code>	Returns a list of the Multi-Region Access Points currently associated with the specified Outpost
<code>list_regional_buckets</code>	Returns a list of all Outposts buckets in an Outpost that are owned by the specified user
<code>list_storage_lens_configurations</code>	Gets a list of Amazon S3 Storage Lens configurations
<code>put_access_point_configuration_for_object_lambda</code>	Replaces configuration for an Object Lambda Access Point
<code>put_access_point_policy</code>	Associates an access policy with the specified access point
<code>put_access_point_policy_for_object_lambda</code>	Creates or replaces resource policy for an Object Lambda Access Point
<code>put_bucket_lifecycle_configuration</code>	This action puts a lifecycle configuration to an Amazon S3 on Outposts bucket
<code>put_bucket_policy</code>	This action puts a bucket policy to an Amazon S3 on Outposts bucket
<code>put_bucket_tagging</code>	This action puts tags on an Amazon S3 on Outposts bucket
<code>put_job_tagging</code>	Sets the supplied tag-set on an S3 Batch Operations job
<code>put_multi_region_access_point_policy</code>	Associates an access control policy with the specified Multi-Region Access Point
<code>put_public_access_block</code>	Creates or modifies the PublicAccessBlock configuration for an Amazon Web Services resource
<code>put_storage_lens_configuration</code>	Puts an Amazon S3 Storage Lens configuration
<code>put_storage_lens_configuration_tagging</code>	Put or replace tags on an existing Amazon S3 Storage Lens configuration
<code>update_job_priority</code>	Updates an existing S3 Batch Operations job's priority
<code>update_job_status</code>	Updates the status for the specified job

Examples

```
## Not run:
svc <- s3control()
svc$create_access_point(
  Foo = 123
)

## End(Not run)
```

s3outposts

Amazon S3 on Outposts

Description

Amazon S3 on Outposts provides access to S3 on Outposts operations.

Usage

```
s3outposts(config = list())
```

Arguments

config	Optional configuration of credentials, endpoint, and/or region. <ul style="list-style-type: none"> • access_key_id: AWS access key ID • secret_access_key: AWS secret access key • session_token: AWS temporary session token • profile: The name of a profile to use. If not given, then the default profile is used. • anonymous: Set anonymous credentials. • endpoint: The complete URL to use for the constructed client. • region: The AWS Region used in instantiating the client. • close_connection: Immediately close all HTTP connections. • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds. • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e., <code>http://s3.amazonaws.com/BUCKET/KEY</code>.
--------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- s3outposts(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
```

```

    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)

```

Operations

create_endpoint	Creates an endpoint and associates it with the specified Outpost
delete_endpoint	Deletes an endpoint
list_endpoints	Lists endpoints associated with the specified Outpost
list_shared_endpoints	Lists all endpoints associated with an Outpost that has been shared by Amazon Web Services Resource

Examples

```

## Not run:
svc <- s3outposts()
svc$create_endpoint(
  Foo = 123
)

## End(Not run)

```

storagegateway

AWS Storage Gateway

Description

Storage Gateway Service

Storage Gateway is the service that connects an on-premises software appliance with cloud-based storage to provide seamless and secure integration between an organization's on-premises IT environment and the Amazon Web Services storage infrastructure. The service enables you to securely upload data to the Amazon Web Services Cloud for cost effective backup and rapid disaster recovery.

Use the following links to get started using the *Storage Gateway Service API Reference*:

- **Storage Gateway required request headers:** Describes the required headers that you must send with every POST request to Storage Gateway.
- **Signing requests:** Storage Gateway requires that you authenticate every request you send; this topic describes how sign such a request.

- **Error responses:** Provides reference information about Storage Gateway errors.
- **Operations in Storage Gateway:** Contains detailed descriptions of all Storage Gateway operations, their request parameters, response elements, possible errors, and examples of requests and responses.
- **Storage Gateway endpoints and quotas:** Provides a list of each Amazon Web Services Region and the endpoints available for use with Storage Gateway.

Storage Gateway resource IDs are in uppercase. When you use these resource IDs with the Amazon EC2 API, EC2 expects resource IDs in lowercase. You must change your resource ID to lowercase to use it with the EC2 API. For example, in Storage Gateway the ID for a volume might be `vol-AA22BB012345DAF670`. When you use this ID with the EC2 API, you must change it to `vol-aa22bb012345daf670`. Otherwise, the EC2 API might not behave as expected.

IDs for Storage Gateway volumes and Amazon EBS snapshots created from gateway volumes are changing to a longer format. Starting in December 2016, all new volumes and snapshots will be created with a 17-character string. Starting in April 2016, you will be able to use these longer IDs so you can test your systems with the new format. For more information, see [Longer EC2 and EBS resource IDs](#).

For example, a volume Amazon Resource Name (ARN) with the longer volume ID format looks like the following:

```
arn:aws:storagegateway:us-west-2:111122223333:gateway/sgw-12A3456B/volume/vol-1122AABCCDDEEFFG.
```

A snapshot ID with the longer ID format looks like the following: `snap-78e226633445566ee`.

For more information, see [Announcement: Heads-up – Longer Storage Gateway volume and snapshot IDs coming in 2016](#).

Usage

```
storagegateway(config = list())
```

Arguments

<code>config</code>	Optional configuration of credentials, endpoint, and/or region. <ul style="list-style-type: none"> • access_key_id: AWS access key ID • secret_access_key: AWS secret access key • session_token: AWS temporary session token • profile: The name of a profile to use. If not given, then the default profile is used. • anonymous: Set anonymous credentials. • endpoint: The complete URL to use for the constructed client. • region: The AWS Region used in instantiating the client. • close_connection: Immediately close all HTTP connections. • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds. • s3_force_path_style: Set this to <code>true</code> to force the request to use path-style addressing, i.e., <code>http://s3.amazonaws.com/BUCKET/KEY</code>.
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- storagegateway(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical"
  )
)
```

Operations

activate_gateway	Activates the gateway you previously deployed on your host
add_cache	Configures one or more gateway local disks as cache for a gateway
add_tags_to_resource	Adds one or more tags to the specified resource
add_upload_buffer	Configures one or more gateway local disks as upload buffer for a specified gateway
add_working_storage	Configures one or more gateway local disks as working storage for a gateway
assign_tape_pool	Assigns a tape to a tape pool for archiving
associate_file_system	Associate an Amazon FSx file system with the FSx File Gateway
attach_volume	Connects a volume to an iSCSI connection and then attaches the volume to the gateway
cancel_archival	Cancels archiving of a virtual tape to the virtual tape shelf (VTS) after the archiving is complete
cancel_retrieval	Cancels retrieval of a virtual tape from the virtual tape shelf (VTS) to a gateway
create_cachedi_scsi_volume	Creates a cached volume on a specified cached volume gateway
create_nfs_file_share	Creates a Network File System (NFS) file share on an existing S3 File Gateway
create_smb_file_share	Creates a Server Message Block (SMB) file share on an existing S3 File Gateway
create_snapshot	Initiates a snapshot of a volume
create_snapshot_from_volume_recovery_point	Initiates a snapshot of a gateway from a volume recovery point
create_storedi_scsi_volume	Creates a volume on a specified gateway
create_tape_pool	Creates a new custom tape pool
create_tapes	Creates one or more virtual tapes
create_tape_with_barcode	Creates a virtual tape by using your own barcode
delete_automatic_tape_creation_policy	Deletes the automatic tape creation policy of a gateway

<code>delete_bandwidth_rate_limit</code>	Deletes the bandwidth rate limits of a gateway
<code>delete_chap_credentials</code>	Deletes Challenge-Handshake Authentication Protocol (CHAP) credentials
<code>delete_file_share</code>	Deletes a file share from an S3 File Gateway
<code>delete_gateway</code>	Deletes a gateway
<code>delete_snapshot_schedule</code>	Deletes a snapshot of a volume
<code>delete_tape</code>	Deletes the specified virtual tape
<code>delete_tape_archive</code>	Deletes the specified virtual tape from the virtual tape shelf (VTS)
<code>delete_tape_pool</code>	Delete a custom tape pool
<code>delete_volume</code>	Deletes the specified storage volume that you previously created using the C
<code>describe_availability_monitor_test</code>	Returns information about the most recent high availability monitoring test t
<code>describe_bandwidth_rate_limit</code>	Returns the bandwidth rate limits of a gateway
<code>describe_bandwidth_rate_limit_schedule</code>	Returns information about the bandwidth rate limit schedule of a gateway
<code>describe_cache</code>	Returns information about the cache of a gateway
<code>describe_cachedi_scsi_volumes</code>	Returns a description of the gateway volumes specified in the request
<code>describe_chap_credentials</code>	Returns an array of Challenge-Handshake Authentication Protocol (CHAP)
<code>describe_file_system_associations</code>	Gets the file system association information
<code>describe_gateway_information</code>	Returns metadata about a gateway such as its name, network interfaces, conf
<code>describe_maintenance_start_time</code>	Returns your gateway's weekly maintenance start time including the day and
<code>describe_nfs_file_shares</code>	Gets a description for one or more Network File System (NFS) file shares fr
<code>describe_smb_file_shares</code>	Gets a description for one or more Server Message Block (SMB) file shares
<code>describe_smb_settings</code>	Gets a description of a Server Message Block (SMB) file share settings from
<code>describe_snapshot_schedule</code>	Describes the snapshot schedule for the specified gateway volume
<code>describe_storedi_scsi_volumes</code>	Returns the description of the gateway volumes specified in the request
<code>describe_tape_archives</code>	Returns a description of specified virtual tapes in the virtual tape shelf (VTS)
<code>describe_tape_recovery_points</code>	Returns a list of virtual tape recovery points that are available for the specifi
<code>describe_tapes</code>	Returns a description of the specified Amazon Resource Name (ARN) of vir
<code>describe_upload_buffer</code>	Returns information about the upload buffer of a gateway
<code>describe_vtl_devices</code>	Returns a description of virtual tape library (VTL) devices for the specified t
<code>describe_working_storage</code>	Returns information about the working storage of a gateway
<code>detach_volume</code>	Disconnects a volume from an iSCSI connection and then detaches the volu
<code>disable_gateway</code>	Disables a tape gateway when the gateway is no longer functioning
<code>disassociate_file_system</code>	Disassociates an Amazon FSx file system from the specified gateway
<code>join_domain</code>	Adds a file gateway to an Active Directory domain
<code>list_automatic_tape_creation_policies</code>	Lists the automatic tape creation policies for a gateway
<code>list_file_shares</code>	Gets a list of the file shares for a specific S3 File Gateway, or the list of file s
<code>list_file_system_associations</code>	Gets a list of FileSystemAssociationSummary objects
<code>list_gateways</code>	Lists gateways owned by an Amazon Web Services account in an Amazon V
<code>list_local_disks</code>	Returns a list of the gateway's local disks
<code>list_tags_for_resource</code>	Lists the tags that have been added to the specified resource
<code>list_tape_pools</code>	Lists custom tape pools
<code>list_tapes</code>	Lists virtual tapes in your virtual tape library (VTL) and your virtual tape sh
<code>list_volume_initiators</code>	Lists iSCSI initiators that are connected to a volume
<code>list_volume_recovery_points</code>	Lists the recovery points for a specified gateway
<code>list_volumes</code>	Lists the iSCSI stored volumes of a gateway
<code>notify_when_uploaded</code>	Sends you notification through CloudWatch Events when all files written to
<code>refresh_cache</code>	Refreshes the cached inventory of objects for the specified file share
<code>remove_tags_from_resource</code>	Removes one or more tags from the specified resource
<code>reset_cache</code>	Resets all cache disks that have encountered an error and makes the disks av

<code>retrieve_tape_archive</code>	Retrieves an archived virtual tape from the virtual tape shelf (VTS) to a tape
<code>retrieve_tape_recovery_point</code>	Retrieves the recovery point for the specified virtual tape
<code>set_local_console_password</code>	Sets the password for your VM local console
<code>set_smb_guest_password</code>	Sets the password for the guest user smbguest
<code>shutdown_gateway</code>	Shuts down a gateway
<code>start_availability_monitor_test</code>	Start a test that verifies that the specified gateway is configured for High Availability
<code>start_gateway</code>	Starts a gateway that you previously shut down (see ShutdownGateway)
<code>update_automatic_tape_creation_policy</code>	Updates the automatic tape creation policy of a gateway
<code>update_bandwidth_rate_limit</code>	Updates the bandwidth rate limits of a gateway
<code>update_bandwidth_rate_limit_schedule</code>	Updates the bandwidth rate limit schedule for a specified gateway
<code>update_chap_credentials</code>	Updates the Challenge-Handshake Authentication Protocol (CHAP) credentials
<code>update_file_system_association</code>	Updates a file system association
<code>update_gateway_information</code>	Updates a gateway's metadata, which includes the gateway's name and time zone
<code>update_gateway_software_now</code>	Updates the gateway virtual machine (VM) software
<code>update_maintenance_start_time</code>	Updates a gateway's weekly maintenance start time information, including duration
<code>update_nfs_file_share</code>	Updates a Network File System (NFS) file share
<code>update_smb_file_share</code>	Updates a Server Message Block (SMB) file share
<code>update_smb_file_share_visibility</code>	Controls whether the shares on an S3 File Gateway are visible in a net view
<code>update_smb_local_groups</code>	Updates the list of Active Directory users and groups that have special permissions
<code>update_smb_security_strategy</code>	Updates the SMB security strategy on a file gateway
<code>update_snapshot_schedule</code>	Updates a snapshot schedule configured for a gateway volume
<code>update_vtl_device_type</code>	Updates the type of medium changer in a tape gateway

Examples

```
## Not run:
svc <- storagegateway()
# Activates the gateway you previously deployed on your host.
svc$activate_gateway(
  ActivationKey = "29AV1-30FV9-VVIUB-NKT0I-LR06V",
  GatewayName = "My_Gateway",
  GatewayRegion = "us-east-1",
  GatewayTimezone = "GMT-12:00",
  GatewayType = "STORED",
  MediumChangerType = "AWS-Gateway-VTL",
  TapeDriveType = "IBM-ULT3580-TD5"
)

## End(Not run)
```

Index

abort_multipart_upload, [19](#), [23](#)
abort_vault_lock, [19](#)
activate_gateway, [32](#)
add_cache, [32](#)
add_tags_to_resource, [32](#)
add_tags_to_vault, [19](#)
add_upload_buffer, [32](#)
add_working_storage, [32](#)
assign_tape_pool, [32](#)
associate_file_system, [32](#)
associate_file_system_aliases, [16](#)
associate_user_to_permission_group, [14](#)
attach_volume, [32](#)

backup, [2](#)
backupstorage, [5](#)

cancel_archival, [32](#)
cancel_data_repository_task, [16](#)
cancel_retrieval, [32](#)
complete_multipart_upload, [19](#), [23](#)
complete_snapshot, [10](#)
complete_vault_lock, [19](#)
copy_backup, [16](#)
copy_object, [23](#)
create_access_point, [12](#), [27](#)
create_access_point_for_object_lambda, [27](#)
create_backup, [16](#)
create_backup_plan, [3](#)
create_backup_selection, [4](#)
create_backup_vault, [4](#)
create_bucket, [23](#), [27](#)
create_cachedi_scsi_volume, [32](#)
create_changeset, [14](#)
create_data_repository_association, [16](#)
create_data_repository_task, [16](#)
create_data_view, [14](#)
create_dataset, [14](#)
create_endpoint, [30](#)
create_file_system, [12](#), [16](#)
create_file_system_from_backup, [16](#)
create_framework, [4](#)
create_job, [27](#)
create_lifecycle_policy, [8](#)
create_mount_target, [12](#)
create_multi_region_access_point, [27](#)
create_multipart_upload, [23](#)
create_nfs_file_share, [32](#)
create_permission_group, [14](#)
create_replication_configuration, [12](#)
create_report_plan, [4](#)
create_rule, [22](#)
create_smb_file_share, [32](#)
create_snapshot, [16](#), [32](#)
create_snapshot_from_volume_recovery_point, [32](#)
create_storage_virtual_machine, [16](#)
create_storedi_scsi_volume, [32](#)
create_tags, [12](#)
create_tape_pool, [32](#)
create_tape_with_barcode, [32](#)
create_tapes, [32](#)
create_user, [14](#)
create_vault, [19](#)
create_volume, [16](#)
create_volume_from_backup, [16](#)

delete_access_point, [12](#), [27](#)
delete_access_point_for_object_lambda, [27](#)
delete_access_point_policy, [27](#)
delete_access_point_policy_for_object_lambda, [27](#)
delete_archive, [19](#)
delete_automatic_tape_creation_policy, [32](#)
delete_backup, [17](#)
delete_backup_plan, [4](#)
delete_backup_selection, [4](#)

- delete_backup_vault, [4](#)
- delete_backup_vault_access_policy, [4](#)
- delete_backup_vault_lock_configuration, [4](#)
- delete_backup_vault_notifications, [4](#)
- delete_bandwidth_rate_limit, [33](#)
- delete_bucket, [23](#), [27](#)
- delete_bucket_analytics_configuration, [23](#)
- delete_bucket_cors, [23](#)
- delete_bucket_encryption, [23](#)
- delete_bucket_intelligent_tiering_configuration, [23](#)
- delete_bucket_inventory_configuration, [23](#)
- delete_bucket_lifecycle, [23](#)
- delete_bucket_lifecycle_configuration, [27](#)
- delete_bucket_metrics_configuration, [23](#)
- delete_bucket_ownership_controls, [24](#)
- delete_bucket_policy, [24](#), [27](#)
- delete_bucket_replication, [24](#)
- delete_bucket_tagging, [24](#), [27](#)
- delete_bucket_website, [24](#)
- delete_chap_credentials, [33](#)
- delete_data_repository_association, [17](#)
- delete_dataset, [14](#)
- delete_endpoint, [30](#)
- delete_file_share, [33](#)
- delete_file_system, [12](#), [17](#)
- delete_file_system_policy, [12](#)
- delete_framework, [4](#)
- delete_gateway, [33](#)
- delete_job_tagging, [27](#)
- delete_lifecycle_policy, [8](#)
- delete_mount_target, [12](#)
- delete_multi_region_access_point, [27](#)
- delete_object, [6](#), [24](#)
- delete_object_tagging, [24](#)
- delete_objects, [24](#)
- delete_permission_group, [14](#)
- delete_public_access_block, [24](#), [27](#)
- delete_recovery_point, [4](#)
- delete_replication_configuration, [12](#)
- delete_report_plan, [4](#)
- delete_rule, [22](#)
- delete_snapshot, [17](#)
- delete_snapshot_schedule, [33](#)
- delete_storage_lens_configuration, [27](#)
- delete_storage_lens_configuration_tagging, [27](#)
- delete_storage_virtual_machine, [17](#)
- delete_tags, [12](#)
- delete_tape, [33](#)
- delete_tape_archive, [33](#)
- delete_tape_pool, [33](#)
- delete_vault, [19](#)
- delete_vault_access_policy, [19](#)
- delete_vault_notifications, [19](#)
- delete_volume, [17](#), [33](#)
- describe_access_points, [12](#)
- describe_account_preferences, [12](#)
- describe_availability_monitor_test, [33](#)
- describe_backup_job, [4](#)
- describe_backup_policy, [12](#)
- describe_backup_vault, [4](#)
- describe_backups, [17](#)
- describe_bandwidth_rate_limit, [33](#)
- describe_bandwidth_rate_limit_schedule, [33](#)
- describe_cache, [33](#)
- describe_cached_i_scsi_volumes, [33](#)
- describe_chap_credentials, [33](#)
- describe_copy_job, [4](#)
- describe_data_repository_associations, [17](#)
- describe_data_repository_tasks, [17](#)
- describe_file_system_aliases, [17](#)
- describe_file_system_associations, [33](#)
- describe_file_system_policy, [12](#)
- describe_file_systems, [12](#), [17](#)
- describe_framework, [4](#)
- describe_gateway_information, [33](#)
- describe_global_settings, [4](#)
- describe_job, [19](#), [27](#)
- describe_lifecycle_configuration, [12](#)
- describe_maintenance_start_time, [33](#)
- describe_mount_target_security_groups, [12](#)
- describe_mount_targets, [12](#)
- describe_multi_region_access_point_operation, [27](#)
- describe_nfs_file_shares, [33](#)
- describe_protected_resource, [4](#)
- describe_recovery_point, [4](#)

- describe_region_settings, [4](#)
- describe_replication_configurations, [12](#)
- describe_report_job, [4](#)
- describe_report_plan, [4](#)
- describe_restore_job, [4](#)
- describe_smb_file_shares, [33](#)
- describe_smb_settings, [33](#)
- describe_snapshot_schedule, [33](#)
- describe_snapshots, [17](#)
- describe_storage_virtual_machines, [17](#)
- describe_storedi_scsi_volumes, [33](#)
- describe_tags, [12](#)
- describe_tape_archives, [33](#)
- describe_tape_recovery_points, [33](#)
- describe_tapes, [33](#)
- describe_upload_buffer, [33](#)
- describe_vault, [19](#)
- describe_volumes, [17](#)
- describe_vtl_devices, [33](#)
- describe_working_storage, [33](#)
- detach_volume, [33](#)
- disable_gateway, [33](#)
- disable_user, [14](#)
- disassociate_file_system, [33](#)
- disassociate_file_system_aliases, [17](#)
- disassociate_recovery_point, [4](#)
- disassociate_user_from_permission_group, [14](#)
- d1m, [7](#)
- download_file, [24](#)
- ebs, [9](#)
- efs, [11](#)
- enable_user, [14](#)
- export_backup_plan_template, [4](#)
- finspacedata, [13](#)
- fsx, [15](#)
- generate_presigned_url, [24](#)
- get_access_point, [27](#)
- get_access_point_configuration_for_object_lambda, [27](#)
- get_access_point_for_object_lambda, [27](#)
- get_access_point_policy, [27](#)
- get_access_point_policy_for_object_lambda, [27](#)
- get_access_point_policy_status, [27](#)
- get_access_point_policy_status_for_object_lambda, [28](#)
- get_backup_plan, [4](#)
- get_backup_plan_from_json, [4](#)
- get_backup_plan_from_template, [4](#)
- get_backup_selection, [4](#)
- get_backup_vault_access_policy, [4](#)
- get_backup_vault_notifications, [4](#)
- get_bucket, [28](#)
- get_bucket_accelerate_configuration, [24](#)
- get_bucket_acl, [24](#)
- get_bucket_analytics_configuration, [24](#)
- get_bucket_cors, [24](#)
- get_bucket_encryption, [24](#)
- get_bucket_intelligent_tiering_configuration, [24](#)
- get_bucket_inventory_configuration, [24](#)
- get_bucket_lifecycle, [24](#)
- get_bucket_lifecycle_configuration, [24, 28](#)
- get_bucket_location, [24](#)
- get_bucket_logging, [24](#)
- get_bucket_metrics_configuration, [24](#)
- get_bucket_notification, [24](#)
- get_bucket_notification_configuration, [24](#)
- get_bucket_ownership_controls, [24](#)
- get_bucket_policy, [24, 28](#)
- get_bucket_policy_status, [24](#)
- get_bucket_replication, [24](#)
- get_bucket_request_payment, [24](#)
- get_bucket_tagging, [24, 28](#)
- get_bucket_versioning, [24](#)
- get_bucket_website, [24](#)
- get_changeset, [14](#)
- get_chunk, [6](#)
- get_data_retrieval_policy, [19](#)
- get_data_view, [14](#)
- get_dataset, [14](#)
- get_external_data_view_access_details, [14](#)
- get_job_output, [19](#)
- get_job_tagging, [28](#)
- get_lifecycle_policies, [8](#)
- get_lifecycle_policy, [8](#)
- get_multi_region_access_point, [28](#)
- get_multi_region_access_point_policy, [28](#)

- 28
- get_multi_region_access_point_policy_status, 28
- get_object, 24
- get_object_acl, 24
- get_object_attributes, 24
- get_object_legal_hold, 24
- get_object_lock_configuration, 24
- get_object_metadata, 6
- get_object_retention, 24
- get_object_tagging, 24
- get_object_torrent, 24
- get_permission_group, 14
- get_programmatic_access_credentials, 14
- get_public_access_block, 24, 28
- get_recovery_point_restore_metadata, 4
- get_rule, 22
- get_snapshot_block, 10
- get_storage_lens_configuration, 28
- get_storage_lens_configuration_tagging, 28
- get_supported_resource_types, 4
- get_user, 14
- get_vault_access_policy, 19
- get_vault_lock, 19
- get_vault_notifications, 19
- get_working_location, 14
- glacier, 17

- head_bucket, 24
- head_object, 24

- initiate_job, 19
- initiate_multipart_upload, 19
- initiate_vault_lock, 19

- join_domain, 33

- list_access_points, 28
- list_access_points_for_object_lambda, 28
- list_automatic_tape_creation_policies, 33
- list_backup_jobs, 4
- list_backup_plan_templates, 4
- list_backup_plan_versions, 4
- list_backup_plans, 4
- list_backup_selections, 4
- list_backup_vaults, 4
- list_bucket_analytics_configurations, 24
- list_bucket_intelligent_tiering_configurations, 24
- list_bucket_inventory_configurations, 24
- list_bucket_metrics_configurations, 24
- list_buckets, 25
- list_changed_blocks, 10
- list_changesets, 14
- list_chunks, 6
- list_copy_jobs, 4
- list_data_views, 15
- list_datasets, 15
- list_endpoints, 30
- list_file_shares, 33
- list_file_system_associations, 33
- list_frameworks, 4
- list_gateways, 33
- list_jobs, 20, 28
- list_local_disks, 33
- list_multi_region_access_points, 28
- list_multipart_uploads, 20, 25
- list_object_versions, 25
- list_objects, 6, 25
- list_objects_v2, 25
- list_parts, 20, 25
- list_permission_groups, 15
- list_permission_groups_by_user, 15
- list_protected_resources, 4
- list_provisioned_capacity, 20
- list_recovery_points_by_backup_vault, 4
- list_recovery_points_by_resource, 4
- list_regional_buckets, 28
- list_report_jobs, 4
- list_report_plans, 4
- list_restore_jobs, 4
- list_rules, 22
- list_shared_endpoints, 30
- list_snapshot_blocks, 10
- list_storage_lens_configurations, 28
- list_tags, 5
- list_tags_for_resource, 8, 12, 17, 22, 33
- list_tags_for_vault, 20
- list_tape_pools, 33
- list_tapes, 33

- list_users, [15](#)
- list_users_by_permission_group, [15](#)
- list_vaults, [20](#)
- list_volume_initiators, [33](#)
- list_volume_recovery_points, [33](#)
- list_volumes, [33](#)

- modify_mount_target_security_groups, [12](#)

- notify_object_complete, [6](#)
- notify_when_uploaded, [33](#)

- purchase_provisioned_capacity, [20](#)
- put_access_point_configuration_for_object_lambda, [28](#)
- put_access_point_policy, [28](#)
- put_access_point_policy_for_object_lambda, [28](#)
- put_account_preferences, [12](#)
- put_backup_policy, [12](#)
- put_backup_vault_access_policy, [5](#)
- put_backup_vault_lock_configuration, [5](#)
- put_backup_vault_notifications, [5](#)
- put_bucket_accelerate_configuration, [25](#)
- put_bucket_acl, [25](#)
- put_bucket_analytics_configuration, [25](#)
- put_bucket_cors, [25](#)
- put_bucket_encryption, [25](#)
- put_bucket_intelligent_tiering_configuration, [25](#)
- put_bucket_inventory_configuration, [25](#)
- put_bucket_lifecycle, [25](#)
- put_bucket_lifecycle_configuration, [25, 28](#)
- put_bucket_logging, [25](#)
- put_bucket_metrics_configuration, [25](#)
- put_bucket_notification, [25](#)
- put_bucket_notification_configuration, [25](#)
- put_bucket_ownership_controls, [25](#)
- put_bucket_policy, [25, 28](#)
- put_bucket_replication, [25](#)
- put_bucket_request_payment, [25](#)
- put_bucket_tagging, [25, 28](#)
- put_bucket_versioning, [25](#)
- put_bucket_website, [25](#)
- put_chunk, [6](#)
- put_file_system_policy, [12](#)
- put_job_tagging, [28](#)
- put_lifecycle_configuration, [12](#)
- put_multi_region_access_point_policy, [28](#)
- put_object, [7, 25](#)
- put_object_acl, [25](#)
- put_object_legal_hold, [25](#)
- put_object_lock_configuration, [25](#)
- put_object_retention, [25](#)
- put_object_tagging, [25](#)
- put_public_access_block, [25, 28](#)
- put_snapshot_block, [10](#)
- put_storage_lens_configuration, [28](#)
- put_storage_lens_configuration_tagging, [28](#)

- recyclebin, [20](#)
- refresh_cache, [33](#)
- release_file_system_nfs_v3_locks, [17](#)
- remove_tags_from_resource, [33](#)
- remove_tags_from_vault, [20](#)
- reset_cache, [33](#)
- reset_user_password, [15](#)
- restore_object, [25](#)
- restore_volume_from_snapshot, [17](#)
- retrieve_tape_archive, [34](#)
- retrieve_tape_recovery_point, [34](#)

- s3, [22](#)
- s3control, [26](#)
- s3outposts, [29](#)
- select_object_content, [25](#)
- set_data_retrieval_policy, [20](#)
- set_local_console_password, [34](#)
- set_smb_guest_password, [34](#)
- set_vault_access_policy, [20](#)
- set_vault_notifications, [20](#)
- shutdown_gateway, [34](#)
- start_availability_monitor_test, [34](#)
- start_backup_job, [5](#)
- start_copy_job, [5](#)
- start_gateway, [34](#)
- start_object, [7](#)
- start_report_job, [5](#)
- start_restore_job, [5](#)
- start_snapshot, [10](#)
- stop_backup_job, [5](#)
- storagegateway, [30](#)

tag_resource, [5](#), [8](#), [12](#), [17](#), [22](#)

untag_resource, [5](#), [8](#), [12](#), [17](#), [22](#)

update_automatic_tape_creation_policy, [34](#)

update_backup_plan, [5](#)

update_bandwidth_rate_limit, [34](#)

update_bandwidth_rate_limit_schedule, [34](#)

update_changeset, [15](#)

update_chap_credentials, [34](#)

update_data_repository_association, [17](#)

update_dataset, [15](#)

update_file_system, [12](#), [17](#)

update_file_system_association, [34](#)

update_framework, [5](#)

update_gateway_information, [34](#)

update_gateway_software_now, [34](#)

update_global_settings, [5](#)

update_job_priority, [28](#)

update_job_status, [28](#)

update_lifecycle_policy, [8](#)

update_maintenance_start_time, [34](#)

update_nfs_file_share, [34](#)

update_permission_group, [15](#)

update_recovery_point_lifecycle, [5](#)

update_region_settings, [5](#)

update_report_plan, [5](#)

update_rule, [22](#)

update_smb_file_share, [34](#)

update_smb_file_share_visibility, [34](#)

update_smb_local_groups, [34](#)

update_smb_security_strategy, [34](#)

update_snapshot, [17](#)

update_snapshot_schedule, [34](#)

update_storage_virtual_machine, [17](#)

update_user, [15](#)

update_volume, [17](#)

update_vtl_device_type, [34](#)

upload_archive, [20](#)

upload_multipart_part, [20](#)

upload_part, [25](#)

upload_part_copy, [25](#)

write_get_object_response, [25](#)