Package ‘paws.storage’

August 3, 2020

Title Amazon Web Services Storage Services

Version 0.1.9

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 (>= 0.3.0)

Suggests testthat

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Collate 'backup_service.R' 'backup_interfaces.R' 'backup_operations.R' 'dlm_service.R' 'dlm_interfaces.R' 'dlm_operations.R' 'efs_service.R' 'efs_interfaces.R' 'efs_operations.R' 'fsx_service.R' 'fsx_interfaces.R' 'fsx_operations.R' 'glacier_service.R' 'glacier_interfaces.R' 'glacier_operations.R' 's3_service.R' 's3_interfaces.R' 's3_operations.R' 's3control_service.R' 's3control_interfaces.R' 's3control_operations.R' 'storagegateway_service.R' 'storagegateway_interfaces.R' 'storagegateway_operations.R'

NeedsCompilation no

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

Maintainer David Kretch <david.kretch@gmail.com>

Repository CRAN

Date/Publication 2020-08-03 09:30:02 UTC

R topics documented:

backup *AWS Backup*

Description

AWS Backup is a unified backup service designed to protect AWS services and their associated data. AWS 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.

Service syntax

```
svc <- backup(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

create_backup_plan	Backup plans are documents that contain information that AWS Backup uses to sche
create_backup_selection	Creates a JSON document that specifies a set of resources to assign to a backup plan
create_backup_vault	Creates a logical container where backups are stored
delete_backup_plan	Deletes a backup plan
delete_backup_selection	Deletes the resource selection associated with a backup plan that is specified by the S
delete_backup_vault	Deletes the backup vault identified by its name
delete_backup_vault_access_policy	Deletes the policy document that manages permissions on a backup vault
delete_backup_vault_notifications	Deletes event notifications for the specified backup vault
delete_recovery_point	Deletes the recovery point specified by a recovery point ID
describe_backup_job	Returns metadata associated with creating a backup of a resource
describe_backup_vault	Returns metadata about a backup vault specified by its name
describe_copy_job	Returns metadata associated with creating a copy of a resource
describe_protected_resource	Returns information about a saved resource, including the last time it was backed up
describe_recovery_point	Returns metadata associated with a recovery point, including ID, status, encryption, a
describe_region_settings	Returns the current service opt-in settings for the Region
describe_restore_job	Returns metadata associated with a restore job that is specified by a job ID
export_backup_plan_template	Returns the backup plan that is specified by the plan ID as a backup template
get_backup_plan	Returns the body of a backup plan in JSON format, in addition to plan metadata
get_backup_plan_from_json	Returns a valid JSON document specifying a backup plan or an error
get_backup_plan_from_template	Returns the template specified by its templateId as a backup plan
get_backup_selection	Returns selection metadata and a document in JSON format that specifies a list of res
get_backup_vault_access_policy	Returns the access policy document that is associated with the named backup vault
get_backup_vault_notifications	Returns event notifications for the specified backup vault
get_recovery_point_restore_metadata	Returns a set of metadata key-value pairs that were used to create the backup
get_supported_resource_types	Returns the AWS resource types supported by AWS Backup
list_backup_jobs	Returns metadata about your backup jobs
list_backup_plans	Returns metadata of your saved backup plans, including Amazon Resource Names (A
list_backup_plan_templates	Returns metadata of your saved backup plan templates, including the template ID, na
list_backup_plan_versions	Returns version metadata of your backup plans, including Amazon Resource Names
list_backup_selections	Returns an array containing metadata of the resources associated with the target back
list_backup_vaults	Returns a list of recovery point storage containers along with information about them
list_copy_jobs	Returns metadata about your copy jobs
list_protected_resources	Returns an array of resources successfully backed up by AWS Backup, including the
list_recovery_points_by_backup_vault	Returns detailed information about the recovery points stored in a backup vault
list_recovery_points_by_resource	Returns detailed information about recovery points of the type specified by a resourc
list_restore_jobs	Returns a list of jobs that AWS Backup initiated to restore a saved resource, includin
list_tags	Returns a list of key-value pairs assigned to a target recovery point, backup plan, or b
put_backup_vault_access_policy	Sets a resource-based policy that is used to manage access permissions on the target
put_backup_vault_notifications	Turns on notifications on a backup vault for the specified topic and events
start_backup_job	Starts a job to create a one-time backup of the specified resource
start_copy_job	Starts a job to create a one-time copy of the specified resource
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 ide
untag_resource	Removes a set of key-value pairs from a recovery point, backup plan, or backup vaul
update_backup_plan	Replaces the body of a saved backup plan identified by its backupPlanId with the inp
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

Examples

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

## End(Not run)
```

---

dlm *Amazon Data Lifecycle Manager*

Description

With Amazon Data Lifecycle Manager, you can manage the lifecycle of your AWS 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

```
dlm(config = list())
```

Arguments

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

Service syntax

```
svc <- dlm(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

create_lifecycle_policy	Creates a policy to manage the lifecycle of the specified AWS 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)
```

`efs` *Amazon Elastic File System*

Description

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

Usage

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

Arguments

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

Service syntax

```
svc <- efs(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

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_tags	Creates or overwrites tags associated with a file system
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_tags	Deletes the specified tags from a file system
describe_access_points	Returns the description of a specific Amazon EFS access point if the AccessPointId
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 syste
describe_lifecycle_configuration	Returns the current LifecycleConfiguration object for the specified Amazon EFS fil
describe_mount_targets	Returns the descriptions of all the current mount targets, or a specific mount target,
describe_mount_target_security_groups	Returns the security groups currently in effect for a mount target
describe_tags	Returns the tags associated with a file system
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_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	Enables lifecycle management by creating a new LifecycleConfiguration object
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 existin

Examples

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

## 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.

Service syntax

```
svc <- fsx(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

cancel_data_repository_task	Cancels an existing Amazon FSx for Lustre data repository task if that task is in either the
create_backup	Creates a backup of an existing Amazon FSx file system
create_data_repository_task	Creates an Amazon FSx for Lustre data repository task
create_file_system	Creates a new, empty Amazon FSx file system
create_file_system_from_backup	Creates a new Amazon FSx file system from an existing Amazon FSx backup
delete_backup	Deletes an Amazon FSx backup, deleting its contents
delete_file_system	Deletes a file system, deleting its contents
describe_backups	Returns the description of specific Amazon FSx backups, if a BackupIds value is provided
describe_data_repository_tasks	Returns the description of specific Amazon FSx for Lustre data repository tasks, if one or
describe_file_systems	Returns the description of specific Amazon FSx file systems, if a FileSystemIds value is pr
list_tags_for_resource	Lists tags for an Amazon FSx file systems and backups in the case of Amazon FSx for Wi
tag_resource	Tags an Amazon FSx resource
untag_resource	This action removes a tag from an Amazon FSx resource
update_file_system	Use this operation to update the configuration of an existing Amazon FSx file system

Examples

```
## Not run:
svc <- fsx()
# This operation creates a new backup.
svc$create_backup(
  FileSystemId = "fs-0498eed5fe91001ec",
  Tags = list(
    list(
      Key = "Name",
      Value = "MyBackup"
    )
  )
)

## End(Not run)
```

glacier *Amazon Glacier*

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.

Service syntax

```
svc <- glacier(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

abort_multipart_upload	This operation aborts a multipart upload identified by the upload ID
abort_vault_lock	This operation aborts the vault locking process if the vault lock is not in the Locked state
add_tags_to_vault	This operation adds the specified tags to a vault
complete_multipart_upload	You call this operation to inform Amazon S3 Glacier (Glacier) that all the archive parts have
complete_vault_lock	This operation completes the vault locking process by transitioning the vault lock from the I
create_vault	This operation creates a new vault with the specified name
delete_archive	This operation deletes an archive from a vault
delete_vault	This operation deletes a vault
delete_vault_access_policy	This operation deletes the access policy associated with the specified vault
delete_vault_notifications	This operation deletes the notification configuration set for a vault
describe_job	This operation returns information about a job you previously initiated, including the job ini
describe_vault	This operation returns information about a vault, including the vault's Amazon Resource Na
get_data_retrieval_policy	This operation returns the current data retrieval policy for the account and region specified i
get_job_output	This operation downloads the output of the job you initiated using InitiateJob
get_vault_access_policy	This operation retrieves the access-policy subresource set on the vault; for more information
get_vault_lock	This operation retrieves the following attributes from the lock-policy subresource set on the
get_vault_notifications	This operation retrieves the notification-configuration subresource of the specified vault
initiate_job	This operation initiates a job of the specified type, which can be a select, an archival retrieva
initiate_multipart_upload	This operation initiates a multipart upload
initiate_vault_lock	This operation initiates the vault locking process by doing the following: - Installing a vault
list_jobs	This operation lists jobs for a vault, including jobs that are in-progress and jobs that have rec
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 vau
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-OssZtLqyFu7sY1_lR7vgFuJV...",
  vaultName = "my-vault"
)
```

```
## 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.

Service syntax

```
svc <- s3(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

abort_multipart_upload	This operation aborts a multipart upload
complete_multipart_upload	Completes a multipart upload by assembling previously uploaded parts
copy_object	Creates a copy of an object that is already stored in Amazon S3
create_bucket	Creates a new bucket
create_multipart_upload	This operation initiates a multipart upload and returns an upload ID
delete_bucket	Deletes the bucket
delete_bucket_analytics_configuration	Deletes an analytics configuration for the bucket (specified by the analytics configur
delete_bucket_cors	Deletes the cors configuration information set for the bucket
delete_bucket_encryption	This implementation of the DELETE operation removes default encryption from the

delete_bucket_inventory_configuration	Deletes an inventory configuration (identified by the inventory ID) from the bucket
delete_bucket_lifecycle	Deletes the lifecycle configuration from the specified bucket
delete_bucket_metrics_configuration	Deletes a metrics configuration for the Amazon CloudWatch request metrics (specifi
delete_bucket_policy	This implementation of the DELETE operation uses the policy subresource to delete
delete_bucket_replication	Deletes the replication configuration from the bucket
delete_bucket_tagging	Deletes the tags from the bucket
delete_bucket_website	This operation removes the website configuration for a bucket
delete_object	Removes the null version (if there is one) of an object and inserts a delete marker, w
delete_objects	This operation enables you to delete multiple objects from a bucket using a single H
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
get_bucket_accelerate_configuration	This implementation of the GET operation uses the accelerate subresource to return
get_bucket_acl	This implementation of the GET operation uses the acl subresource to return the acc
get_bucket_analytics_configuration	This implementation of the GET operation returns an analytics configuration (identi
get_bucket_cors	Returns the cors configuration information set for the bucket
get_bucket_encryption	Returns the default encryption configuration for an Amazon S3 bucket
get_bucket_inventory_configuration	Returns an inventory configuration (identified by the inventory configuration ID) fro
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 using an obje
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 view and m
get_bucket_metrics_configuration	Gets a metrics configuration (specified by the metrics configuration ID) from the buc
get_bucket_notification	No longer used, see GetBucketNotificationConfiguration
get_bucket_notification_configuration	Returns the notification configuration of a 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 bucket i
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_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	Return torrent files from a bucket
get_public_access_block	Retrieves the PublicAccessBlock configuration for an Amazon S3 bucket
head_bucket	This operation is useful to determine if a bucket exists and you have permission to ac
head_object	The HEAD operation retrieves metadata from an object without returning the object
list_bucket_analytics_configurations	Lists the analytics configurations for the 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 operation 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

list_object_versions	Returns metadata about all of the versions of 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 configuratio
put_bucket_cors	Sets the cors configuration for your bucket
put_bucket_encryption	This implementation of the PUT operation uses the encryption subresource to set the
put_bucket_inventory_configuration	This implementation of the PUT operation adds an inventory configuration (identifie
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 lifecycle
put_bucket_logging	Set the logging parameters for a bucket and to specify permissions for who can view
put_bucket_metrics_configuration	Sets a metrics configuration (specified by the metrics configuration ID) for the bucke
put_bucket_notification	No longer used, see the PutBucketNotificationConfiguration operation
put_bucket_notification_configuration	Enables notifications of specified events for a 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 an obje
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 This operation perform
select_object_content	This operation filters the contents of an Amazon S3 object based on a simple structu
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

Examples

```
## Not run:
svc <- s3()
# The following example aborts a multipart upload.
svc$abort_multipart_upload(
  Bucket = "examplebucket",
  Key = "bigobject",
 UploadId = "xadcOB_7YPBOJuoFiQ9cz4P3Pe6FIZwO4f7wN93uHsNBEw97pl5eNwzExg0LAT2dUN91cOmrEQHDsP..."
)

## End(Not run)
```

s3control *AWS S3 Control*

Description

AWS S3 Control provides access to Amazon S3 control plane operations.

Usage

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

Arguments

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

Service syntax

```
svc <- s3control(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

create_access_point	Creates an access point and associates it with the specified bucket
create_job	You can use Amazon S3 Batch Operations to perform large-scale Batch Operations on Ama
delete_access_point	Deletes the specified access point
delete_access_point_policy	Deletes the access point policy for the specified access point
delete_job_tagging	Removes the entire tag set from the specified Amazon S3 Batch Operations job
delete_public_access_block	Removes the PublicAccessBlock configuration for an Amazon Web Services account
describe_job	Retrieves the configuration parameters and status for a Batch Operations job
get_access_point	Returns configuration information about the specified access point
get_access_point_policy	Returns the access point policy associated with the specified access point
get_access_point_policy_status	Indicates whether the specified access point currently has a policy that allows public access
get_job_tagging	Returns the tags on an Amazon S3 Batch Operations job
get_public_access_block	Retrieves the PublicAccessBlock configuration for an Amazon Web Services account
list_access_points	Returns a list of the access points currently associated with the specified bucket

list_jobs	Lists current Amazon S3 Batch Operations jobs and jobs that have ended within the last 30 (
put_access_point_policy	Associates an access policy with the specified access point
put_job_tagging	Set the supplied tag-set on an Amazon S3 Batch Operations job
put_public_access_block	Creates or modifies the PublicAccessBlock configuration for an Amazon Web Services acco
update_job_priority	Updates an existing Amazon S3 Batch Operations job's priority
update_job_status	Updates the status for the specified job

Examples

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

## End(Not run)
```

storagegateway *AWS Storage Gateway*

Description

AWS Storage Gateway Service

AWS 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 AWS storage infrastructure. The service enables you to securely upload data to the AWS Cloud for cost effective backup and rapid disaster recovery.

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

- AWS Storage Gateway required request headers: Describes the required headers that you must send with every POST request to AWS Storage Gateway.
- Signing requests: AWS Storage Gateway requires that you authenticate every request you send; this topic describes how sign such a request.
- Error responses: Provides reference information about AWS Storage Gateway errors.
- Operations in AWS Storage Gateway: Contains detailed descriptions of all AWS Storage Gateway operations, their request parameters, response elements, possible errors, and examples of requests and responses.
- AWS Storage Gateway endpoints and quotas: Provides a list of each AWS Region and the endpoints available for use with AWS Storage Gateway.

AWS 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-1122AABBCCDDEEFFG.

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

For more information, see Announcement: Heads-up – Longer AWS Storage Gateway volume and snapshot IDs coming in 2016.

Usage

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

Arguments

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

Service syntax

```
svc <- storagegateway(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

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

add_working_storage	Configures one or more gateway local disks as working storage for a gatewa
assign_tape_pool	Assigns a tape to a tape pool for archiving
attach_volume	Connects a volume to an iSCSI connection and then attaches the volume to t
cancel_archival	Cancels archiving of a virtual tape to the virtual tape shelf (VTS) after the ar
cancel_retrieval	Cancels retrieval of a virtual tape from the virtual tape shelf (VTS) to a gatev
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 file gateway
create_smb_file_share	Creates a Server Message Block (SMB) file share on an existing 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_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
delete_bandwidth_rate_limit	Deletes the bandwidth rate limits of a gateway
delete_chap_credentials	Deletes Challenge-Handshake Authentication Protocol (CHAP) credentials f
delete_file_share	Deletes a file share from a file gateway
delete_gateway	Deletes a gateway
delete_snapshot_schedule	Deletes a snapshot of a volume
delete_tape	Deletes the specified virtual tape
delete_tape_archive	Deletes the specified virtual tape from the virtual tape shelf (VTS)
delete_volume	Deletes the specified storage volume that you previously created using the C
describe_availability_monitor_test	Returns information about the most recent High Availability monitoring test
describe_bandwidth_rate_limit	Returns the bandwidth rate limits of a gateway
describe_cache	Returns information about the cache of a gateway
describe_cachedi_scsi_volumes	Returns a description of the gateway volumes specified in the request
describe_chap_credentials	Returns an array of Challenge-Handshake Authentication Protocol (CHAP) (
describe_gateway_information	Returns metadata about a gateway such as its name, network interfaces, conf
describe_maintenance_start_time	Returns your gateway's weekly maintenance start time including the day anc
describe_nfs_file_shares	Gets a description for one or more Network File System (NFS) file shares fr
describe_smb_file_shares	Gets a description for one or more Server Message Block (SMB) file shares
describe_smb_settings	Gets a description of a Server Message Block (SMB) file share settings from
describe_snapshot_schedule	Describes the snapshot schedule for the specified gateway volume
describe_storedi_scsi_volumes	Returns the description of the gateway volumes specified in the request
describe_tape_archives	Returns a description of specified virtual tapes in the virtual tape shelf (VTS
describe_tape_recovery_points	Returns a list of virtual tape recovery points that are available for the specifie
describe_tapes	Returns a description of the specified Amazon Resource Name (ARN) of vir
describe_upload_buffer	Returns information about the upload buffer of a gateway
describe_vtl_devices	Returns a description of virtual tape library (VTL) devices for the specified t
describe_working_storage	Returns information about the working storage of a gateway
detach_volume	Disconnects a volume from an iSCSI connection and then detaches the volur
disable_gateway	Disables a tape gateway when the gateway is no longer functioning
join_domain	Adds a file gateway to an Active Directory domain
list_automatic_tape_creation_policies	Lists the automatic tape creation policies for a gateway
list_file_shares	Gets a list of the file shares for a specific file gateway, or the list of file share
list_gateways	Lists gateways owned by an AWS account in an AWS Region specified in th
list_local_disks	Returns a list of the gateway's local disks
list_tags_for_resource	Lists the tags that have been added to the specified resource

list_tapes	Lists virtual tapes in your virtual tape library (VTL) and your virtual tape sh
list_volume_initiators	Lists iSCSI initiators that are connected to a volume
list_volume_recovery_points	Lists the recovery points for a specified gateway
list_volumes	Lists the iSCSI stored volumes of a gateway
notify_when_uploaded	Sends you notification through CloudWatch Events when all files written to
refresh_cache	Refreshes the cache for the specified file share
remove_tags_from_resource	Removes one or more tags from the specified resource
reset_cache	Resets all cache disks that have encountered an error and makes the disks av
retrieve_tape_archive	Retrieves an archived virtual tape from the virtual tape shelf (VTS) to a tape
retrieve_tape_recovery_point	Retrieves the recovery point for the specified virtual tape
set_local_console_password	Sets the password for your VM local console
set_smb_guest_password	Sets the password for the guest user smbguest
shutdown_gateway	Shuts down a gateway
start_availability_monitor_test	Start a test that verifies that the specified gateway is configured for High Ava
start_gateway	Starts a gateway that you previously shut down (see ShutdownGateway)
update_automatic_tape_creation_policy	Updates the automatic tape creation policy of a gateway
update_bandwidth_rate_limit	Updates the bandwidth rate limits of a gateway
update_chap_credentials	Updates the Challenge-Handshake Authentication Protocol (CHAP) credent
update_gateway_information	Updates a gateway's metadata, which includes the gateway's name and time
update_gateway_software_now	Updates the gateway virtual machine (VM) software
update_maintenance_start_time	Updates a gateway's weekly maintenance start time information, including d
update_nfs_file_share	Updates a Network File System (NFS) file share
update_smb_file_share	Updates a Server Message Block (SMB) file share
update_smb_security_strategy	Updates the SMB security strategy on a file gateway
update_snapshot_schedule	Updates a snapshot schedule configured for a gateway volume
update_vtl_device_type	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 = "29AV1-3OFV9-VVIUB-NKT0I-LRO6V",
  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