Package 'paws.security.identity'

August 3, 2020

Title Amazon Web Services Security, Identity, & Compliance Services

Version 0.1.9

Description Interface to Amazon Web Services security, identity, and compliance services, including the 'Identity & Access Management' ('IAM') service for managing access to services and resources, 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 'acm_service.R' 'acm_interfaces.R' 'acm_operations.R' 'acmpca_service.R' 'acmpca_interfaces.R' 'acmpca_operations.R' 'clouddirectory_service.R' 'clouddirectory_interfaces.R' 'clouddirectory_operations.R' 'cloudhsm_service.R' 'cloudhsm_interfaces.R' 'cloudhsm_operations.R' 'cloudhsmv2_service.R' 'cloudhsmv2_interfaces.R' 'cloudhsmv2_operations.R' 'cognitoidentity_service.R' 'cognitoidentity_interfaces.R' 'cognitoidentity_operations.R' 'cognitoidentityprovider_service.R' 'cognitoidentityprovider_interfaces.R' 'cognitoidentityprovider_operations.R' 'cognitosync_service.R' 'cognitosync_interfaces.R' 'cognitosync_operations.R' 'directoryservice_service.R' 'directoryservice_interfaces.R' 'directoryservice_operations.R' 'fms_service.R' 'fms_interfaces.R' 'fms_operations.R' 'guardduty_service.R' 'guardduty_interfaces.R' 'guardduty_operations.R' 'iam_service.R' 'iam_interfaces.R' 'iam_operations.R' 'inspector_service.R' 'inspector_interfaces.R'

'inspector_operations.R' 'kms_service.R' 'kms_interfaces.R'
'kms_operations.R' 'macie_service.R' 'macie_interfaces.R'
'macie_operations.R' 'ram_service.R' 'ram_interfaces.R'
'ram_operations.R' 'secretsmanager_service.R'
'secretsmanager_interfaces.R' 'secretsmanager_operations.R'
'securityhub_service.R' 'securityhub_interfaces.R'
'securityhub_operations.R' 'shield_service.R'
'shield_interfaces.R' 'shield_operations.R' 'sts_service.R'
'sts_interfaces.R' 'sts_operations.R' 'waf_service.R'
'waf_interfaces.R' 'waf_operations.R' 'wafregional_service.R'
'wafregional_interfaces.R' 'wafregional_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 09:30:09 UTC

R topics documented:

acm *AWS Certificate Manager*

Description

Welcome to the AWS Certificate Manager (ACM) API documentation.

You can use ACM to manage SSL/TLS certificates for your AWS-based websites and applications. For general information about using ACM, see the *AWS Certificate Manager User Guide* .

Usage

```
acm(config = list())
```

Arguments

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

Service syntax

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

Operations

add_tags_to_certificate	Adds one or more tags to an ACM certificate
delete_certificate	Deletes a certificate and its associated private key
describe_certificate	Returns detailed metadata about the specified ACM certificate
export_certificate	Exports a private certificate issued by a private certificate authority (CA) for use anywhere
get_certificate	Retrieves an Amazon-issued certificate and its certificate chain
import_certificate	Imports a certificate into AWS Certificate Manager (ACM) to use with services that are integr
list_certificates	Retrieves a list of certificate ARNs and domain names
list_tags_for_certificate	Lists the tags that have been applied to the ACM certificate
remove_tags_from_certificate	Remove one or more tags from an ACM certificate
renew_certificate	Renews an eligable ACM certificate
request_certificate	Requests an ACM certificate for use with other AWS services

resend_validation_email	Resends the email that requests domain ownership validation
update_certificate_options	Updates a certificate

Examples

```
## Not run:
svc <- acm()
svc$add_tags_to_certificate(
  Foo = 123
)

## End(Not run)
```

---

acmpca *AWS Certificate Manager Private Certificate Authority*

Description

This is the *ACM Private CA API Reference*. It provides descriptions, syntax, and usage examples for each of the actions and data types involved in creating and managing private certificate authorities (CA) for your organization.

The documentation for each action shows the Query API request parameters and the XML response. Alternatively, you can use one of the AWS SDKs to access an API that\'s tailored to the programming language or platform that you\'re using. For more information, see AWS SDKs.

Each ACM Private CA API action has a throttling limit which determines the number of times the action can be called per second. For more information, see API Rate Limits in ACM Private CA in the ACM Private CA user guide.

Usage

```
acmpca(config = list())
```

Arguments

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

Service syntax

```
svc <- acmpca(
  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_certificate_authority	Creates a root or subordinate private certificate authority (CA)
create_certificate_authority_audit_report	Creates an audit report that lists every time that your CA private key is used
create_permission	Assigns permissions from a private CA to a designated AWS service
delete_certificate_authority	Deletes a private certificate authority (CA)
delete_permission	Revokes permissions that a private CA assigned to a designated AWS service
describe_certificate_authority	Lists information about your private certificate authority (CA)
describe_certificate_authority_audit_report	Lists information about a specific audit report created by calling the CreateCerti
get_certificate	Retrieves a certificate from your private CA
get_certificate_authority_certificate	Retrieves the certificate and certificate chain for your private certificate authority
get_certificate_authority_csr	Retrieves the certificate signing request (CSR) for your private certificate author
import_certificate_authority_certificate	Imports a signed private CA certificate into ACM Private CA
issue_certificate	Uses your private certificate authority (CA) to issue a client certificate
list_certificate_authorities	Lists the private certificate authorities that you created by using the CreateCertif
list_permissions	Lists all the permissions, if any, that have been assigned by a private CA
list_tags	Lists the tags, if any, that are associated with your private CA
restore_certificate_authority	Restores a certificate authority (CA) that is in the DELETED state
revoke_certificate	Revokes a certificate that was issued inside ACM Private CA
tag_certificate_authority	Adds one or more tags to your private CA
untag_certificate_authority	Remove one or more tags from your private CA
update_certificate_authority	Updates the status or configuration of a private certificate authority (CA)

Examples

```
## Not run:
svc <- acmpca()
svc$create_certificate_authority(
  Foo = 123
)

## End(Not run)
```

clouddirectory *Amazon CloudDirectory*

Description

Amazon Cloud Directory

Amazon Cloud Directory is a component of the AWS Directory Service that simplifies the development and management of cloud-scale web, mobile, and IoT applications. This guide describes the Cloud Directory operations that you can call programmatically and includes detailed information on data types and errors. For information about Cloud Directory features, see AWS Directory Service and the Amazon Cloud Directory Developer Guide.

Usage

```
clouddirectory(config = list())
```

Arguments

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

Service syntax

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

Operations

add_facet_to_object	Adds a new Facet to an object
apply_schema	Copies the input published schema, at the specified version, into the Directory with the sa
attach_object	Attaches an existing object to another object
attach_policy	Attaches a policy object to a regular object
attach_to_index	Attaches the specified object to the specified index
attach_typed_link	Attaches a typed link to a specified source and target object
batch_read	Performs all the read operations in a batch
batch_write	Performs all the write operations in a batch

create_directory	Creates a Directory by copying the published schema into the directory
create_facet	Creates a new Facet in a schema
create_index	Creates an index object
create_object	Creates an object in a Directory
create_schema	Creates a new schema in a development state
create_typed_link_facet	Creates a TypedLinkFacet
delete_directory	Deletes a directory
delete_facet	Deletes a given Facet
delete_object	Deletes an object and its associated attributes
delete_schema	Deletes a given schema
delete_typed_link_facet	Deletes a TypedLinkFacet
detach_from_index	Detaches the specified object from the specified index
detach_object	Detaches a given object from the parent object
detach_policy	Detaches a policy from an object
detach_typed_link	Detaches a typed link from a specified source and target object
disable_directory	Disables the specified directory
enable_directory	Enables the specified directory
get_applied_schema_version	Returns current applied schema version ARN, including the minor version in use
get_directory	Retrieves metadata about a directory
get_facet	Gets details of the Facet, such as facet name, attributes, Rules, or ObjectType
get_link_attributes	Retrieves attributes that are associated with a typed link
get_object_attributes	Retrieves attributes within a facet that are associated with an object
get_object_information	Retrieves metadata about an object
get_schema_as_json	Retrieves a JSON representation of the schema
get_typed_link_facet_information	Returns the identity attribute order for a specific TypedLinkFacet
list_applied_schema_arns	Lists schema major versions applied to a directory
list_attached_indices	Lists indices attached to the specified object
list_development_schema_arns	Retrieves each Amazon Resource Name (ARN) of schemas in the development state
list_directories	Lists directories created within an account
list_facet_attributes	Retrieves attributes attached to the facet
list_facet_names	Retrieves the names of facets that exist in a schema
list_incoming_typed_links	Returns a paginated list of all the incoming TypedLinkSpecifier information for an object
list_index	Lists objects attached to the specified index
list_managed_schema_arns	Lists the major version families of each managed schema
list_object_attributes	Lists all attributes that are associated with an object
list_object_children	Returns a paginated list of child objects that are associated with a given object
list_object_parent_paths	Retrieves all available parent paths for any object type such as node, leaf node, policy noc
list_object_parents	Lists parent objects that are associated with a given object in pagination fashion
list_object_policies	Returns policies attached to an object in pagination fashion
list_outgoing_typed_links	Returns a paginated list of all the outgoing TypedLinkSpecifier information for an object
list_policy_attachments	Returns all of the ObjectIdentifiers to which a given policy is attached
list_published_schema_arns	Lists the major version families of each published schema
list_tags_for_resource	Returns tags for a resource
list_typed_link_facet_attributes	Returns a paginated list of all attribute definitions for a particular TypedLinkFacet
list_typed_link_facet_names	Returns a paginated list of TypedLink facet names for a particular schema
lookup_policy	Lists all policies from the root of the Directory to the object specified
publish_schema	Publishes a development schema with a major version and a recommended minor version
put_schema_from_json	Allows a schema to be updated using JSON upload

remove_facet_from_object	Removes the specified facet from the specified object
tag_resource	An API operation for adding tags to a resource
untag_resource	An API operation for removing tags from a resource
update_facet	Does the following: 1
update_link_attributes	Updates a given typed link's attributes
update_object_attributes	Updates a given object's attributes
update_schema	Updates the schema name with a new name
update_typed_link_facet	Updates a TypedLinkFacet
upgrade_applied_schema	Upgrades a single directory in-place using the PublishedSchemaArn with schema updates
upgrade_published_schema	Upgrades a published schema under a new minor version revision using the current conte

Examples

```
## Not run:
svc <- clouddirectory()
svc$add_facet_to_object(
  Foo = 123
)

## End(Not run)
```

`cloudhsm` *Amazon CloudHSM*

Description

AWS CloudHSM Service

This is documentation for **AWS CloudHSM Classic**. For more information, see AWS CloudHSM Classic FAQs, the AWS CloudHSM Classic User Guide, and the AWS CloudHSM Classic API Reference.

For information about the current version of AWS CloudHSM, see AWS CloudHSM, the AWS CloudHSM User Guide, and the AWS CloudHSM API Reference.

Usage

```
cloudhsm(config = list())
```

Arguments

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

Service syntax

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

Operations

add_tags_to_resource	This is documentation for AWS CLOUDHSM CLASSIC
create_hapg	This is documentation for AWS CLOUDHSM CLASSIC
create_hsm	This is documentation for AWS CLOUDHSM CLASSIC
create_luna_client	This is documentation for AWS CLOUDHSM CLASSIC
delete_hapg	This is documentation for AWS CLOUDHSM CLASSIC
delete_hsm	This is documentation for AWS CLOUDHSM CLASSIC
delete_luna_client	This is documentation for AWS CLOUDHSM CLASSIC
describe_hapg	This is documentation for AWS CLOUDHSM CLASSIC
describe_hsm	This is documentation for AWS CLOUDHSM CLASSIC
describe_luna_client	This is documentation for AWS CLOUDHSM CLASSIC
get_config	This is documentation for AWS CLOUDHSM CLASSIC
list_available_zones	This is documentation for AWS CLOUDHSM CLASSIC
list_hapgs	This is documentation for AWS CLOUDHSM CLASSIC
list_hsms	This is documentation for AWS CLOUDHSM CLASSIC
list_luna_clients	This is documentation for AWS CLOUDHSM CLASSIC
list_tags_for_resource	This is documentation for AWS CLOUDHSM CLASSIC
modify_hapg	This is documentation for AWS CLOUDHSM CLASSIC
modify_hsm	This is documentation for AWS CLOUDHSM CLASSIC
modify_luna_client	This is documentation for AWS CLOUDHSM CLASSIC
remove_tags_from_resource	This is documentation for AWS CLOUDHSM CLASSIC

Examples

```
## Not run:
svc <- cloudhsm()
svc$add_tags_to_resource(
  Foo = 123
)
```

```
## End(Not run)
```

`cloudhsmv2` *AWS CloudHSM V2*

Description

For more information about AWS CloudHSM, see AWS CloudHSM and the AWS CloudHSM User Guide.

Usage

```
cloudhsmv2(config = list())
```

Arguments

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

Service syntax

```
svc <- cloudhsmv2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

copy_backup_to_region	Copy an AWS CloudHSM cluster backup to a different region
create_cluster	Creates a new AWS CloudHSM cluster
create_hsm	Creates a new hardware security module (HSM) in the specified AWS CloudHSM cluster
delete_backup	Deletes a specified AWS CloudHSM backup
delete_cluster	Deletes the specified AWS CloudHSM cluster
delete_hsm	Deletes the specified HSM
describe_backups	Gets information about backups of AWS CloudHSM clusters
describe_clusters	Gets information about AWS CloudHSM clusters

initialize_cluster	Claims an AWS CloudHSM cluster by submitting the cluster certificate issued by your issuing certif
list_tags	Gets a list of tags for the specified AWS CloudHSM cluster
restore_backup	Restores a specified AWS CloudHSM backup that is in the PENDING_DELETION state
tag_resource	Adds or overwrites one or more tags for the specified AWS CloudHSM cluster
untag_resource	Removes the specified tag or tags from the specified AWS CloudHSM cluster

Examples

```
## Not run:
svc <- cloudhsmv2()
svc$copy_backup_to_region(
  Foo = 123
)

## End(Not run)
```

`cognitoidentity` *Amazon Cognito Identity*

Description

Amazon Cognito Federated Identities

Amazon Cognito Federated Identities is a web service that delivers scoped temporary credentials to mobile devices and other untrusted environments. It uniquely identifies a device and supplies the user with a consistent identity over the lifetime of an application.

Using Amazon Cognito Federated Identities, you can enable authentication with one or more third-party identity providers (Facebook, Google, or Login with Amazon) or an Amazon Cognito user pool, and you can also choose to support unauthenticated access from your app. Cognito delivers a unique identifier for each user and acts as an OpenID token provider trusted by AWS Security Token Service (STS) to access temporary, limited-privilege AWS credentials.

For a description of the authentication flow from the Amazon Cognito Developer Guide see Authentication Flow.

For more information see Amazon Cognito Federated Identities.

Usage

```
cognitoidentity(config = list())
```

Arguments

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

Service syntax

```
svc <- cognitoidentity(
  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_identity_pool	Creates a new identity pool
delete_identities	Deletes identities from an identity pool
delete_identity_pool	Deletes an identity pool
describe_identity	Returns metadata related to the given identity, including when the identity was c
describe_identity_pool	Gets details about a particular identity pool, including the pool name, ID descrip
get_credentials_for_identity	Returns credentials for the provided identity ID
get_id	Generates (or retrieves) a Cognito ID
get_identity_pool_roles	Gets the roles for an identity pool
get_open_id_token	Gets an OpenID token, using a known Cognito ID
get_open_id_token_for_developer_identity	Registers (or retrieves) a Cognito IdentityId and an OpenID Connect token for a
list_identities	Lists the identities in an identity pool
list_identity_pools	Lists all of the Cognito identity pools registered for your account
list_tags_for_resource	Lists the tags that are assigned to an Amazon Cognito identity pool
lookup_developer_identity	Retrieves the IdentityID associated with a DeveloperUserIdentifier or the list of
merge_developer_identities	Merges two users having different IdentityIds, existing in the same identity pool
set_identity_pool_roles	Sets the roles for an identity pool
tag_resource	Assigns a set of tags to an Amazon Cognito identity pool
unlink_developer_identity	Unlinks a DeveloperUserIdentifier from an existing identity
unlink_identity	Unlinks a federated identity from an existing account
untag_resource	Removes the specified tags from an Amazon Cognito identity pool
update_identity_pool	Updates an identity pool

Examples

```
## Not run:
svc <- cognitoidentity()
svc$create_identity_pool(
  Foo = 123
```

```
)

## End(Not run)
```

`cognitoidentityprovider` *Amazon Cognito Identity Provider*

Description

Using the Amazon Cognito User Pools API, you can create a user pool to manage directories and users. You can authenticate a user to obtain tokens related to user identity and access policies.

This API reference provides information about user pools in Amazon Cognito User Pools.

For more information, see the Amazon Cognito Documentation.

Usage

```
cognitoidentityprovider(config = list())
```

Arguments

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

Service syntax

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

Operations

add_custom_attributes	Adds additional user attributes to the user pool schema
admin_add_user_to_group	Adds the specified user to the specified group
admin_confirm_sign_up	Confirms user registration as an admin without using a confirmation code
admin_create_user	Creates a new user in the specified user pool

admin_delete_user	Deletes a user as an administrator
admin_delete_user_attributes	Deletes the user attributes in a user pool as an administrator
admin_disable_provider_for_user	Disables the user from signing in with the specified external (SAML or social) identity
admin_disable_user	Disables the specified user
admin_enable_user	Enables the specified user as an administrator
admin_forget_device	Forgets the device, as an administrator
admin_get_device	Gets the device, as an administrator
admin_get_user	Gets the specified user by user name in a user pool as an administrator
admin_initiate_auth	Initiates the authentication flow, as an administrator
admin_link_provider_for_user	Links an existing user account in a user pool (DestinationUser) to an identity from an e
admin_list_devices	Lists devices, as an administrator
admin_list_groups_for_user	Lists the groups that the user belongs to
admin_list_user_auth_events	Lists a history of user activity and any risks detected as part of Amazon Cognito advan
admin_remove_user_from_group	Removes the specified user from the specified group
admin_reset_user_password	Resets the specified user's password in a user pool as an administrator
admin_respond_to_auth_challenge	Responds to an authentication challenge, as an administrator
admin_set_user_mfa_preference	Sets the user's multi-factor authentication (MFA) preference, including which MFA op
admin_set_user_password	Sets the specified user's password in a user pool as an administrator
admin_set_user_settings	_This action is no longer supported
admin_update_auth_event_feedback	Provides feedback for an authentication event as to whether it was from a valid user
admin_update_device_status	Updates the device status as an administrator
admin_update_user_attributes	Updates the specified user's attributes, including developer attributes, as an administrat
admin_user_global_sign_out	Signs out users from all devices, as an administrator
associate_software_token	Returns a unique generated shared secret key code for the user account
change_password	Changes the password for a specified user in a user pool
confirm_device	Confirms tracking of the device
confirm_forgot_password	Allows a user to enter a confirmation code to reset a forgotten password
confirm_sign_up	Confirms registration of a user and handles the existing alias from a previous user
create_group	Creates a new group in the specified user pool
create_identity_provider	Creates an identity provider for a user pool
create_resource_server	Creates a new OAuth2
create_user_import_job	Creates the user import job
create_user_pool	Creates a new Amazon Cognito user pool and sets the password policy for the pool
create_user_pool_client	Creates the user pool client
create_user_pool_domain	Creates a new domain for a user pool
delete_group	Deletes a group
delete_identity_provider	Deletes an identity provider for a user pool
delete_resource_server	Deletes a resource server
delete_user	Allows a user to delete himself or herself
delete_user_attributes	Deletes the attributes for a user
delete_user_pool	Deletes the specified Amazon Cognito user pool
delete_user_pool_client	Allows the developer to delete the user pool client
delete_user_pool_domain	Deletes a domain for a user pool
describe_identity_provider	Gets information about a specific identity provider
describe_resource_server	Describes a resource server
describe_risk_configuration	Describes the risk configuration
describe_user_import_job	Describes the user import job
describe_user_pool	Returns the configuration information and metadata of the specified user pool

describe_user_pool_client	Client method for returning the configuration information and metadata of the specifie
describe_user_pool_domain	Gets information about a domain
forget_device	Forgets the specified device
forgot_password	Calling this API causes a message to be sent to the end user with a confirmation code t
get_csv_header	Gets the header information for the
get_device	Gets the device
get_group	Gets a group
get_identity_provider_by_identifier	Gets the specified identity provider
get_signing_certificate	This method takes a user pool ID, and returns the signing certificate
get_ui_customization	Gets the UI Customization information for a particular app client's app UI, if there is s
get_user	Gets the user attributes and metadata for a user
get_user_attribute_verification_code	Gets the user attribute verification code for the specified attribute name
get_user_pool_mfa_config	Gets the user pool multi-factor authentication (MFA) configuration
global_sign_out	Signs out users from all devices
initiate_auth	Initiates the authentication flow
list_devices	Lists the devices
list_groups	Lists the groups associated with a user pool
list_identity_providers	Lists information about all identity providers for a user pool
list_resource_servers	Lists the resource servers for a user pool
list_tags_for_resource	Lists the tags that are assigned to an Amazon Cognito user pool
list_user_import_jobs	Lists the user import jobs
list_user_pool_clients	Lists the clients that have been created for the specified user pool
list_user_pools	Lists the user pools associated with an AWS account
list_users	Lists the users in the Amazon Cognito user pool
list_users_in_group	Lists the users in the specified group
resend_confirmation_code	Resends the confirmation (for confirmation of registration) to a specific user in the use
respond_to_auth_challenge	Responds to the authentication challenge
set_risk_configuration	Configures actions on detected risks
set_ui_customization	Sets the UI customization information for a user pool's built-in app UI
set_user_mfa_preference	Set the user's multi-factor authentication (MFA) method preference, including which M
set_user_pool_mfa_config	Set the user pool multi-factor authentication (MFA) configuration
set_user_settings	_This action is no longer supported
sign_up	Registers the user in the specified user pool and creates a user name, password, and use
start_user_import_job	Starts the user import
stop_user_import_job	Stops the user import job
tag_resource	Assigns a set of tags to an Amazon Cognito user pool
untag_resource	Removes the specified tags from an Amazon Cognito user pool
update_auth_event_feedback	Provides the feedback for an authentication event whether it was from a valid user or n
update_device_status	Updates the device status
update_group	Updates the specified group with the specified attributes
update_identity_provider	Updates identity provider information for a user pool
update_resource_server	Updates the name and scopes of resource server
update_user_attributes	Allows a user to update a specific attribute (one at a time)
update_user_pool	Updates the specified user pool with the specified attributes
update_user_pool_client	Updates the specified user pool app client with the specified attributes
update_user_pool_domain	Updates the Secure Sockets Layer (SSL) certificate for the custom domain for your use
verify_software_token	Use this API to register a user's entered TOTP code and mark the user's software token
verify_user_attribute	Verifies the specified user attributes in the user pool

Examples

```
## Not run:
svc <- cognitoidentityprovider()
svc$add_custom_attributes(
  Foo = 123
)

## End(Not run)
```

`cognitosync` *Amazon Cognito Sync*

Description

Amazon Cognito Sync provides an AWS service and client library that enable cross-device syncing of application-related user data. High-level client libraries are available for both iOS and Android. You can use these libraries to persist data locally so that it\'s available even if the device is offline. Developer credentials don\'t need to be stored on the mobile device to access the service. You can use Amazon Cognito to obtain a normalized user ID and credentials. User data is persisted in a dataset that can store up to 1 MB of key-value pairs, and you can have up to 20 datasets per user identity.

With Amazon Cognito Sync, the data stored for each identity is accessible only to credentials assigned to that identity. In order to use the Cognito Sync service, you need to make API calls using credentials retrieved with Amazon Cognito Identity service.

If you want to use Cognito Sync in an Android or iOS application, you will probably want to make API calls via the AWS Mobile SDK. To learn more, see the Developer Guide for Android and the Developer Guide for iOS.

Usage

```
cognitosync(config = list())
```

Arguments

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

Service syntax

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

```
      session_token = "string"
    ),
    profile = "string"
  ),
  endpoint = "string",
  region = "string"
 )
)
```

Operations

bulk_publish	Initiates a bulk publish of all existing datasets for an Identity Pool to the configured stream
delete_dataset	Deletes the specific dataset
describe_dataset	Gets meta data about a dataset by identity and dataset name
describe_identity_pool_usage	Gets usage details (for example, data storage) about a particular identity pool
describe_identity_usage	Gets usage information for an identity, including number of datasets and data usage
get_bulk_publish_details	Get the status of the last BulkPublish operation for an identity pool
get_cognito_events	Gets the events and the corresponding Lambda functions associated with an identity pool
get_identity_pool_configuration	Gets the configuration settings of an identity pool
list_datasets	Lists datasets for an identity
list_identity_pool_usage	Gets a list of identity pools registered with Cognito
list_records	Gets paginated records, optionally changed after a particular sync count for a dataset and id
register_device	Registers a device to receive push sync notifications
set_cognito_events	Sets the AWS Lambda function for a given event type for an identity pool
set_identity_pool_configuration	Sets the necessary configuration for push sync
subscribe_to_dataset	Subscribes to receive notifications when a dataset is modified by another device
unsubscribe_from_dataset	Unsubscribes from receiving notifications when a dataset is modified by another device
update_records	Posts updates to records and adds and deletes records for a dataset and user

Examples

```
## Not run:
svc <- cognitosync()
svc$bulk_publish(
  Foo = 123
)

## End(Not run)
```

`directoryservice` *AWS Directory Service*

Description

AWS Directory Service is a web service that makes it easy for you to setup and run directories in the AWS cloud, or connect your AWS resources with an existing on-premises Microsoft Active Directory. This guide provides detailed information about AWS Directory Service operations, data types, parameters, and errors. For information about AWS Directory Services features, see AWS Directory Service and the AWS Directory Service Administration Guide.

AWS provides SDKs that consist of libraries and sample code for various programming languages and platforms (Java, Ruby, .Net, iOS, Android, etc.). The SDKs provide a convenient way to create programmatic access to AWS Directory Service and other AWS services. For more information about the AWS SDKs, including how to download and install them, see Tools for Amazon Web Services.

Usage

```
directoryservice(config = list())
```

Arguments

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

Service syntax

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

Operations

accept_shared_directory	Accepts a directory sharing request that was sent from the directory owner account
add_ip_routes	If the DNS server for your on-premises domain uses a publicly addressable IP addre
add_tags_to_resource	Adds or overwrites one or more tags for the specified directory
cancel_schema_extension	Cancels an in-progress schema extension to a Microsoft AD directory
connect_directory	Creates an AD Connector to connect to an on-premises directory
create_alias	Creates an alias for a directory and assigns the alias to the directory
create_computer	Creates a computer account in the specified directory, and joins the computer to the
create_conditional_forwarder	Creates a conditional forwarder associated with your AWS directory
create_directory	Creates a Simple AD directory

create_log_subscription	Creates a subscription to forward real-time Directory Service domain controller secu
create_microsoft_ad	Creates a Microsoft AD directory in the AWS Cloud
create_snapshot	Creates a snapshot of a Simple AD or Microsoft AD directory in the AWS cloud
create_trust	AWS Directory Service for Microsoft Active Directory allows you to configure trust
delete_conditional_forwarder	Deletes a conditional forwarder that has been set up for your AWS directory
delete_directory	Deletes an AWS Directory Service directory
delete_log_subscription	Deletes the specified log subscription
delete_snapshot	Deletes a directory snapshot
delete_trust	Deletes an existing trust relationship between your AWS Managed Microsoft AD di
deregister_certificate	Deletes from the system the certificate that was registered for a secured LDAP conn
deregister_event_topic	Removes the specified directory as a publisher to the specified SNS topic
describe_certificate	Displays information about the certificate registered for a secured LDAP connection
describe_conditional_forwarders	Obtains information about the conditional forwarders for this account
describe_directories	Obtains information about the directories that belong to this account
describe_domain_controllers	Provides information about any domain controllers in your directory
describe_event_topics	Obtains information about which SNS topics receive status messages from the speci
describe_ldaps_settings	Describes the status of LDAP security for the specified directory
describe_shared_directories	Returns the shared directories in your account
describe_snapshots	Obtains information about the directory snapshots that belong to this account
describe_trusts	Obtains information about the trust relationships for this account
disable_ldaps	Deactivates LDAP secure calls for the specified directory
disable_radius	Disables multi-factor authentication (MFA) with the Remote Authentication Dial In
disable_sso	Disables single-sign on for a directory
enable_ldaps	Activates the switch for the specific directory to always use LDAP secure calls
enable_radius	Enables multi-factor authentication (MFA) with the Remote Authentication Dial In
enable_sso	Enables single sign-on for a directory
get_directory_limits	Obtains directory limit information for the current Region
get_snapshot_limits	Obtains the manual snapshot limits for a directory
list_certificates	For the specified directory, lists all the certificates registered for a secured LDAP co
list_ip_routes	Lists the address blocks that you have added to a directory
list_log_subscriptions	Lists the active log subscriptions for the AWS account
list_schema_extensions	Lists all schema extensions applied to a Microsoft AD Directory
list_tags_for_resource	Lists all tags on a directory
register_certificate	Registers a certificate for secured LDAP connection
register_event_topic	Associates a directory with an SNS topic
reject_shared_directory	Rejects a directory sharing request that was sent from the directory owner account
remove_ip_routes	Removes IP address blocks from a directory
remove_tags_from_resource	Removes tags from a directory
reset_user_password	Resets the password for any user in your AWS Managed Microsoft AD or Simple A
restore_from_snapshot	Restores a directory using an existing directory snapshot
share_directory	Shares a specified directory (DirectoryId) in your AWS account (directory owner) w
start_schema_extension	Applies a schema extension to a Microsoft AD directory
unshare_directory	Stops the directory sharing between the directory owner and consumer accounts
update_conditional_forwarder	Updates a conditional forwarder that has been set up for your AWS directory
update_number_of_domain_controllers	Adds or removes domain controllers to or from the directory
update_radius	Updates the Remote Authentication Dial In User Service (RADIUS) server informa
update_trust	Updates the trust that has been set up between your AWS Managed Microsoft AD d
verify_trust	AWS Directory Service for Microsoft Active Directory allows you to configure and

Examples

```
## Not run:
svc <- directoryservice()
svc$accept_shared_directory(
  Foo = 123
)

## End(Not run)
```

---

fms *Firewall Management Service*

Description

AWS Firewall Manager

This is the *AWS Firewall Manager API Reference*. This guide is for developers who need detailed information about the AWS Firewall Manager API actions, data types, and errors. For detailed information about AWS Firewall Manager features, see the AWS Firewall Manager Developer Guide.

Usage

```
fms(config = list())
```

Arguments

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

Service syntax

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

Operations

associate_admin_account	Sets the AWS Firewall Manager administrator account
delete_notification_channel	Deletes an AWS Firewall Manager association with the IAM role and the Amazon Simple Noti
delete_policy	Permanently deletes an AWS Firewall Manager policy
disassociate_admin_account	Disassociates the account that has been set as the AWS Firewall Manager administrator accoun
get_admin_account	Returns the AWS Organizations master account that is associated with AWS Firewall Manager
get_compliance_detail	Returns detailed compliance information about the specified member account
get_notification_channel	Information about the Amazon Simple Notification Service (SNS) topic that is used to record A
get_policy	Returns information about the specified AWS Firewall Manager policy
get_protection_status	If you created a Shield Advanced policy, returns policy-level attack summary information in th
list_compliance_status	Returns an array of PolicyComplianceStatus objects in the response
list_member_accounts	Returns a MemberAccounts object that lists the member accounts in the administrator's AWS c
list_policies	Returns an array of PolicySummary objects in the response
list_tags_for_resource	Retrieves the list of tags for the specified AWS resource
put_notification_channel	Designates the IAM role and Amazon Simple Notification Service (SNS) topic that AWS Firev
put_policy	Creates an AWS Firewall Manager policy
tag_resource	Adds one or more tags to an AWS resource
untag_resource	Removes one or more tags from an AWS resource

Examples

```
## Not run:
svc <- fms()
svc$associate_admin_account(
  Foo = 123
)

## End(Not run)
```

`guardduty` *Amazon GuardDuty*

Description

Amazon GuardDuty is a continuous security monitoring service that analyzes and processes the following data sources: VPC Flow Logs, AWS CloudTrail event logs, and DNS logs. It uses threat intelligence feeds (such as lists of malicious IPs and domains) and machine learning to identify unexpected, potentially unauthorized, and malicious activity within your AWS environment. This can include issues like escalations of privileges, uses of exposed credentials, or communication with malicious IPs, URLs, or domains. For example, GuardDuty can detect compromised EC2 instances that serve malware or mine bitcoin.

GuardDuty also monitors AWS account access behavior for signs of compromise. Some examples of this are unauthorized infrastructure deployments such as EC2 instances deployed in a Region that has never been used, or unusual API calls like a password policy change to reduce password strength.

GuardDuty informs you of the status of your AWS environment by producing security findings that you can view in the GuardDuty console or through Amazon CloudWatch events. For more information, see the *Amazon GuardDuty User Guide* .

Usage

```
guardduty(config = list())
```

Arguments

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

Service syntax

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

Operations

accept_invitation	Accepts the invitation to be monitored by a master GuardDuty account
archive_findings	Archives GuardDuty findings that are specified by the list of finding IDs
create_detector	Creates a single Amazon GuardDuty detector
create_filter	Creates a filter using the specified finding criteria
create_ip_set	Creates a new IPSet, which is called a trusted IP list in the console user interface
create_members	Creates member accounts of the current AWS account by specifying a list of AWS acc
create_publishing_destination	Creates a publishing destination to export findings to
create_sample_findings	Generates example findings of types specified by the list of finding types
create_threat_intel_set	Creates a new ThreatIntelSet
decline_invitations	Declines invitations sent to the current member account by AWS accounts specified by
delete_detector	Deletes an Amazon GuardDuty detector that is specified by the detector ID
delete_filter	Deletes the filter specified by the filter name
delete_invitations	Deletes invitations sent to the current member account by AWS accounts specified by
delete_ip_set	Deletes the IPSet specified by the ipSetId
delete_members	Deletes GuardDuty member accounts (to the current GuardDuty master account) spec
delete_publishing_destination	Deletes the publishing definition with the specified destinationId
delete_threat_intel_set	Deletes the ThreatIntelSet specified by the ThreatIntelSet ID
describe_organization_configuration	Returns information about the account selected as the delegated administrator for Gua

describe_publishing_destination	Returns information about the publishing destination specified by the provided destina
disable_organization_admin_account	Disables an AWS account within the Organization as the GuardDuty delegated admini
disassociate_from_master_account	Disassociates the current GuardDuty member account from its master account
disassociate_members	Disassociates GuardDuty member accounts (to the current GuardDuty master account
enable_organization_admin_account	Enables an AWS account within the organization as the GuardDuty delegated adminis
get_detector	Retrieves an Amazon GuardDuty detector specified by the detectorId
get_filter	Returns the details of the filter specified by the filter name
get_findings	Describes Amazon GuardDuty findings specified by finding IDs
get_findings_statistics	Lists Amazon GuardDuty findings statistics for the specified detector ID
get_invitations_count	Returns the count of all GuardDuty membership invitations that were sent to the curre
get_ip_set	Retrieves the IPSet specified by the ipSetId
get_master_account	Provides the details for the GuardDuty master account associated with the current Gua
get_members	Retrieves GuardDuty member accounts (to the current GuardDuty master account) spe
get_threat_intel_set	Retrieves the ThreatIntelSet that is specified by the ThreatIntelSet ID
invite_members	Invites other AWS accounts (created as members of the current AWS account by Crea
list_detectors	Lists detectorIds of all the existing Amazon GuardDuty detector resources
list_filters	Returns a paginated list of the current filters
list_findings	Lists Amazon GuardDuty findings for the specified detector ID
list_invitations	Lists all GuardDuty membership invitations that were sent to the current AWS accoun
list_ip_sets	Lists the IPSets of the GuardDuty service specified by the detector ID
list_members	Lists details about all member accounts for the current GuardDuty master account
list_organization_admin_accounts	Lists the accounts configured as GuardDuty delegated administrators
list_publishing_destinations	Returns a list of publishing destinations associated with the specified dectectorId
list_tags_for_resource	Lists tags for a resource
list_threat_intel_sets	Lists the ThreatIntelSets of the GuardDuty service specified by the detector ID
start_monitoring_members	Turns on GuardDuty monitoring of the specified member accounts
stop_monitoring_members	Stops GuardDuty monitoring for the specified member accounts
tag_resource	Adds tags to a resource
unarchive_findings	Unarchives GuardDuty findings specified by the findingIds
untag_resource	Removes tags from a resource
update_detector	Updates the Amazon GuardDuty detector specified by the detectorId
update_filter	Updates the filter specified by the filter name
update_findings_feedback	Marks the specified GuardDuty findings as useful or not useful
update_ip_set	Updates the IPSet specified by the IPSet ID
update_organization_configuration	Updates the delegated administrator account with the values provided
update_publishing_destination	Updates information about the publishing destination specified by the destinationId
update_threat_intel_set	Updates the ThreatIntelSet specified by the ThreatIntelSet ID

Examples

```
## Not run:
svc <- guardduty()
svc$accept_invitation(
  Foo = 123
)

## End(Not run)
```

iam *AWS Identity and Access Management*

Description

AWS Identity and Access Management (IAM) is a web service for securely controlling access to AWS services. With IAM, you can centrally manage users, security credentials such as access keys, and permissions that control which AWS resources users and applications can access. For more information about IAM, see AWS Identity and Access Management (IAM) and the AWS Identity and Access Management User Guide.

Usage

```
iam(config = list())
```

Arguments

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

Service syntax

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

Operations

add_client_id_to_open_id_connect_provider	Adds a new client ID (also known as audience) to the list of client IDs a
add_role_to_instance_profile	Adds the specified IAM role to the specified instance profile
add_user_to_group	Adds the specified user to the specified group
attach_group_policy	Attaches the specified managed policy to the specified IAM group
attach_role_policy	Attaches the specified managed policy to the specified IAM role
attach_user_policy	Attaches the specified managed policy to the specified user
change_password	Changes the password of the IAM user who is calling this operation

create_access_key	Creates a new AWS secret access key and corresponding AWS access k
create_account_alias	Creates an alias for your AWS account
create_group	Creates a new group
create_instance_profile	Creates a new instance profile
create_login_profile	Creates a password for the specified user, giving the user the ability to a
create_open_id_connect_provider	Creates an IAM entity to describe an identity provider (IdP) that suppor
create_policy	Creates a new managed policy for your AWS account
create_policy_version	Creates a new version of the specified managed policy
create_role	Creates a new role for your AWS account
create_saml_provider	Creates an IAM resource that describes an identity provider (IdP) that su
create_service_linked_role	Creates an IAM role that is linked to a specific AWS service
create_service_specific_credential	Generates a set of credentials consisting of a user name and password th
create_user	Creates a new IAM user for your AWS account
create_virtual_mfa_device	Creates a new virtual MFA device for the AWS account
deactivate_mfa_device	Deactivates the specified MFA device and removes it from association w
delete_access_key	Deletes the access key pair associated with the specified IAM user
delete_account_alias	Deletes the specified AWS account alias
delete_account_password_policy	Deletes the password policy for the AWS account
delete_group	Deletes the specified IAM group
delete_group_policy	Deletes the specified inline policy that is embedded in the specified IAM
delete_instance_profile	Deletes the specified instance profile
delete_login_profile	Deletes the password for the specified IAM user, which terminates the u
delete_open_id_connect_provider	Deletes an OpenID Connect identity provider (IdP) resource object in IA
delete_policy	Deletes the specified managed policy
delete_policy_version	Deletes the specified version from the specified managed policy
delete_role	Deletes the specified role
delete_role_permissions_boundary	Deletes the permissions boundary for the specified IAM role
delete_role_policy	Deletes the specified inline policy that is embedded in the specified IAM
delete_saml_provider	Deletes a SAML provider resource in IAM
delete_server_certificate	Deletes the specified server certificate
delete_service_linked_role	Submits a service-linked role deletion request and returns a DeletionTas
delete_service_specific_credential	Deletes the specified service-specific credential
delete_signing_certificate	Deletes a signing certificate associated with the specified IAM user
delete_ssh_public_key	Deletes the specified SSH public key
delete_user	Deletes the specified IAM user
delete_user_permissions_boundary	Deletes the permissions boundary for the specified IAM user
delete_user_policy	Deletes the specified inline policy that is embedded in the specified IAM
delete_virtual_mfa_device	Deletes a virtual MFA device
detach_group_policy	Removes the specified managed policy from the specified IAM group
detach_role_policy	Removes the specified managed policy from the specified role
detach_user_policy	Removes the specified managed policy from the specified user
enable_mfa_device	Enables the specified MFA device and associates it with the specified IA
generate_credential_report	Generates a credential report for the AWS account
generate_organizations_access_report	Generates a report for service last accessed data for AWS Organizations
generate_service_last_accessed_details	Generates a report that includes details about when an IAM resource (us
get_access_key_last_used	Retrieves information about when the specified access key was last used
get_account_authorization_details	Retrieves information about all IAM users, groups, roles, and policies in
get_account_password_policy	Retrieves the password policy for the AWS account

get_account_summary	Retrieves information about IAM entity usage and IAM quotas in the A
get_context_keys_for_custom_policy	Gets a list of all of the context keys referenced in the input policies
get_context_keys_for_principal_policy	Gets a list of all of the context keys referenced in all the IAM policies th
get_credential_report	Retrieves a credential report for the AWS account
get_group	Returns a list of IAM users that are in the specified IAM group
get_group_policy	Retrieves the specified inline policy document that is embedded in the s
get_instance_profile	Retrieves information about the specified instance profile, including the
get_login_profile	Retrieves the user name and password-creation date for the specified IA
get_open_id_connect_provider	Returns information about the specified OpenID Connect (OIDC) provi
get_organizations_access_report	Retrieves the service last accessed data report for AWS Organizations th
get_policy	Retrieves information about the specified managed policy, including the
get_policy_version	Retrieves information about the specified version of the specified manag
get_role	Retrieves information about the specified role, including the role's path,
get_role_policy	Retrieves the specified inline policy document that is embedded with th
get_saml_provider	Returns the SAML provider metadocument that was uploaded when the
get_server_certificate	Retrieves information about the specified server certificate stored in IAM
get_service_last_accessed_details	Retrieves a service last accessed report that was created using the Gener
get_service_last_accessed_details_with_entities	After you generate a group or policy report using the GenerateServiceLa
get_service_linked_role_deletion_status	Retrieves the status of your service-linked role deletion
get_ssh_public_key	Retrieves the specified SSH public key, including metadata about the ke
get_user	Retrieves information about the specified IAM user, including the user's
get_user_policy	Retrieves the specified inline policy document that is embedded in the s
list_access_keys	Returns information about the access key IDs associated with the specif
list_account_aliases	Lists the account alias associated with the AWS account (Note: you can
list_attached_group_policies	Lists all managed policies that are attached to the specified IAM group
list_attached_role_policies	Lists all managed policies that are attached to the specified IAM role
list_attached_user_policies	Lists all managed policies that are attached to the specified IAM user
list_entities_for_policy	Lists all IAM users, groups, and roles that the specified managed policy
list_group_policies	Lists the names of the inline policies that are embedded in the specified
list_groups	Lists the IAM groups that have the specified path prefix
list_groups_for_user	Lists the IAM groups that the specified IAM user belongs to
list_instance_profiles	Lists the instance profiles that have the specified path prefix
list_instance_profiles_for_role	Lists the instance profiles that have the specified associated IAM role
list_mfa_devices	Lists the MFA devices for an IAM user
list_open_id_connect_providers	Lists information about the IAM OpenID Connect (OIDC) provider reso
list_policies	Lists all the managed policies that are available in your AWS account, in
list_policies_granting_service_access	Retrieves a list of policies that the IAM identity (user, group, or role) ca
list_policy_versions	Lists information about the versions of the specified managed policy, inc
list_role_policies	Lists the names of the inline policies that are embedded in the specified
list_roles	Lists the IAM roles that have the specified path prefix
list_role_tags	Lists the tags that are attached to the specified role
list_saml_providers	Lists the SAML provider resource objects defined in IAM in the accoun
list_server_certificates	Lists the server certificates stored in IAM that have the specified path pr
list_service_specific_credentials	Returns information about the service-specific credentials associated wi
list_signing_certificates	Returns information about the signing certificates associated with the sp
list_ssh_public_keys	Returns information about the SSH public keys associated with the spec
list_user_policies	Lists the names of the inline policies embedded in the specified IAM us
list_users	Lists the IAM users that have the specified path prefix

list_user_tags	Lists the tags that are attached to the specified user
list_virtual_mfa_devices	Lists the virtual MFA devices defined in the AWS account by assignmer
put_group_policy	Adds or updates an inline policy document that is embedded in the spec
put_role_permissions_boundary	Adds or updates the policy that is specified as the IAM role's permissio
put_role_policy	Adds or updates an inline policy document that is embedded in the spec
put_user_permissions_boundary	Adds or updates the policy that is specified as the IAM user's permissio
put_user_policy	Adds or updates an inline policy document that is embedded in the spec
remove_client_id_from_open_id_connect_provider	Removes the specified client ID (also known as audience) from the list c
remove_role_from_instance_profile	Removes the specified IAM role from the specified EC2 instance profile
remove_user_from_group	Removes the specified user from the specified group
reset_service_specific_credential	Resets the password for a service-specific credential
resync_mfa_device	Synchronizes the specified MFA device with its IAM resource object on
set_default_policy_version	Sets the specified version of the specified policy as the policy's default (
set_security_token_service_preferences	Sets the specified version of the global endpoint token as the token versi
simulate_custom_policy	Simulate how a set of IAM policies and optionally a resource-based pol
simulate_principal_policy	Simulate how a set of IAM policies attached to an IAM entity works wi
tag_role	Adds one or more tags to an IAM role
tag_user	Adds one or more tags to an IAM user
untag_role	Removes the specified tags from the role
untag_user	Removes the specified tags from the user
update_access_key	Changes the status of the specified access key from Active to Inactive, o
update_account_password_policy	Updates the password policy settings for the AWS account
update_assume_role_policy	Updates the policy that grants an IAM entity permission to assume a rol
update_group	Updates the name and/or the path of the specified IAM group
update_login_profile	Changes the password for the specified IAM user
update_open_id_connect_provider_thumbprint	Replaces the existing list of server certificate thumbprints associated wi
update_role	Updates the description or maximum session duration setting of a role
update_role_description	Use UpdateRole instead
update_saml_provider	Updates the metadata document for an existing SAML provider resourc
update_server_certificate	Updates the name and/or the path of the specified server certificate store
update_service_specific_credential	Sets the status of a service-specific credential to Active or Inactive
update_signing_certificate	Changes the status of the specified user signing certificate from active to
update_ssh_public_key	Sets the status of an IAM user's SSH public key to active or inactive
update_user	Updates the name and/or the path of the specified IAM user
upload_server_certificate	Uploads a server certificate entity for the AWS account
upload_signing_certificate	Uploads an X
upload_ssh_public_key	Uploads an SSH public key and associates it with the specified IAM use

Examples

```
## Not run:
svc <- iam()
# The following add-client-id-to-open-id-connect-provider command adds the
# client ID my-application-ID to the OIDC provider named
# server.example.com:
svc$add_client_id_to_open_id_connect_provider(
  ClientID = "my-application-ID",
```

```
  OpenIDConnectProviderArn = "arn:aws:iam::123456789012:oidc-provider/server.example.com"
)

## End(Not run)
```

`inspector` *Amazon Inspector*

Description

Amazon Inspector enables you to analyze the behavior of your AWS resources and to identify potential security issues. For more information, see Amazon Inspector User Guide.

Usage

```
inspector(config = list())
```

Arguments

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

Service syntax

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

Operations

add_attributes_to_findings	Assigns attributes (key and value pairs) to the findings that are specified by the ARNs o
create_assessment_target	Creates a new assessment target using the ARN of the resource group that is generated
create_assessment_template	Creates an assessment template for the assessment target that is specified by the ARN o
create_exclusions_preview	Starts the generation of an exclusions preview for the specified assessment template
create_resource_group	Creates a resource group using the specified set of tags (key and value pairs) that are us
delete_assessment_run	Deletes the assessment run that is specified by the ARN of the assessment run

delete_assessment_target	Deletes the assessment target that is specified by the ARN of the assessment target
delete_assessment_template	Deletes the assessment template that is specified by the ARN of the assessment templa
describe_assessment_runs	Describes the assessment runs that are specified by the ARNs of the assessment runs
describe_assessment_targets	Describes the assessment targets that are specified by the ARNs of the assessment targ
describe_assessment_templates	Describes the assessment templates that are specified by the ARNs of the assessment te
describe_cross_account_access_role	Describes the IAM role that enables Amazon Inspector to access your AWS account
describe_exclusions	Describes the exclusions that are specified by the exclusions' ARNs
describe_findings	Describes the findings that are specified by the ARNs of the findings
describe_resource_groups	Describes the resource groups that are specified by the ARNs of the resource groups
describe_rules_packages	Describes the rules packages that are specified by the ARNs of the rules packages
get_assessment_report	Produces an assessment report that includes detailed and comprehensive results of a sp
get_exclusions_preview	Retrieves the exclusions preview (a list of ExclusionPreview objects) specified by the p
get_telemetry_metadata	Information about the data that is collected for the specified assessment run
list_assessment_run_agents	Lists the agents of the assessment runs that are specified by the ARNs of the assessmen
list_assessment_runs	Lists the assessment runs that correspond to the assessment templates that are specified
list_assessment_targets	Lists the ARNs of the assessment targets within this AWS account
list_assessment_templates	Lists the assessment templates that correspond to the assessment targets that are specifi
list_event_subscriptions	Lists all the event subscriptions for the assessment template that is specified by the AR
list_exclusions	List exclusions that are generated by the assessment run
list_findings	Lists findings that are generated by the assessment runs that are specified by the ARNs
list_rules_packages	Lists all available Amazon Inspector rules packages
list_tags_for_resource	Lists all tags associated with an assessment template
preview_agents	Previews the agents installed on the EC2 instances that are part of the specified assessn
register_cross_account_access_role	Registers the IAM role that grants Amazon Inspector access to AWS Services needed t
remove_attributes_from_findings	Removes entire attributes (key and value pairs) from the findings that are specified by t
set_tags_for_resource	Sets tags (key and value pairs) to the assessment template that is specified by the ARN
start_assessment_run	Starts the assessment run specified by the ARN of the assessment template
stop_assessment_run	Stops the assessment run that is specified by the ARN of the assessment run
subscribe_to_event	Enables the process of sending Amazon Simple Notification Service (SNS) notification
unsubscribe_from_event	Disables the process of sending Amazon Simple Notification Service (SNS) notificatio
update_assessment_target	Updates the assessment target that is specified by the ARN of the assessment target

Examples

```
## Not run:
svc <- inspector()
# Assigns attributes (key and value pairs) to the findings that are
# specified by the ARNs of the findings.
svc$add_attributes_to_findings(
  attributes = list(
    list(
      key = "Example",
      value = "example"
    )
  ),
  findingArns = list(
   "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-8l1VIE0D/run/0-Z0..."
  )
```

```
)

## End(Not run)
```

---

kms *AWS Key Management Service*

Description

AWS Key Management Service (AWS KMS) is an encryption and key management web service. This guide describes the AWS KMS operations that you can call programmatically. For general information about AWS KMS, see the *AWS Key Management Service Developer Guide* .

AWS provides SDKs that consist of libraries and sample code for various programming languages and platforms (Java, Ruby, .Net, macOS, Android, etc.). The SDKs provide a convenient way to create programmatic access to AWS KMS and other AWS services. For example, the SDKs take care of tasks such as signing requests (see below), managing errors, and retrying requests automatically. For more information about the AWS SDKs, including how to download and install them, see Tools for Amazon Web Services.

We recommend that you use the AWS SDKs to make programmatic API calls to AWS KMS.

Clients must support TLS (Transport Layer Security) 1.0. We recommend TLS 1.2. Clients must also support cipher suites with Perfect Forward Secrecy (PFS) such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems such as Java 7 and later support these modes.

Signing Requests

Requests must be signed by using an access key ID and a secret access key. We strongly recommend that you *do not* use your AWS account (root) access key ID and secret key for everyday work with AWS KMS. Instead, use the access key ID and secret access key for an IAM user. You can also use the AWS Security Token Service to generate temporary security credentials that you can use to sign requests.

All AWS KMS operations require Signature Version 4.

Logging API Requests

AWS KMS supports AWS CloudTrail, a service that logs AWS API calls and related events for your AWS account and delivers them to an Amazon S3 bucket that you specify. By using the information collected by CloudTrail, you can determine what requests were made to AWS KMS, who made the request, when it was made, and so on. To learn more about CloudTrail, including how to turn it on and find your log files, see the AWS CloudTrail User Guide.

Additional Resources

For more information about credentials and request signing, see the following:

- AWS Security Credentials - This topic provides general information about the types of credentials used for accessing AWS.
- Temporary Security Credentials - This section of the *IAM User Guide* describes how to create and use temporary security credentials.

- Signature Version 4 Signing Process - This set of topics walks you through the process of signing a request using an access key ID and a secret access key.

Commonly Used API Operations

Of the API operations discussed in this guide, the following will prove the most useful for most applications. You will likely perform operations other than these, such as creating keys and assigning policies, by using the console.

- Encrypt
- Decrypt
- GenerateDataKey
- GenerateDataKeyWithoutPlaintext

Usage

```
kms(config = list())
```

Arguments

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

Service syntax

```
svc <- kms(
  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_key_deletion	Cancels the deletion of a customer master key (CMK)
connect_custom_key_store	Connects or reconnects a custom key store to its associated AWS CloudHSM clu
create_alias	Creates a display name for a customer managed customer master key (CMK)
create_custom_key_store	Creates a custom key store that is associated with an AWS CloudHSM cluster tha
create_grant	Adds a grant to a customer master key (CMK)
create_key	Creates a unique customer managed customer master key (CMK) in your AWS a
decrypt	Decrypts ciphertext that was encrypted by a AWS KMS customer master key (Cl
delete_alias	Deletes the specified alias

delete_custom_key_store	Deletes a custom key store
delete_imported_key_material	Deletes key material that you previously imported
describe_custom_key_stores	Gets information about custom key stores in the account and region
describe_key	Provides detailed information about a customer master key (CMK)
disable_key	Sets the state of a customer master key (CMK) to disabled, thereby preventing its
disable_key_rotation	Disables automatic rotation of the key material for the specified symmetric custo
disconnect_custom_key_store	Disconnects the custom key store from its associated AWS CloudHSM cluster
enable_key	Sets the key state of a customer master key (CMK) to enabled
enable_key_rotation	Enables automatic rotation of the key material for the specified symmetric custor
encrypt	Encrypts plaintext into ciphertext by using a customer master key (CMK)
generate_data_key	Generates a unique symmetric data key for client-side encryption
generate_data_key_pair	Generates a unique asymmetric data key pair
generate_data_key_pair_without_plaintext	Generates a unique asymmetric data key pair
generate_data_key_without_plaintext	Generates a unique symmetric data key
generate_random	Returns a random byte string that is cryptographically secure
get_key_policy	Gets a key policy attached to the specified customer master key (CMK)
get_key_rotation_status	Gets a Boolean value that indicates whether automatic rotation of the key materia
get_parameters_for_import	Returns the items you need to import key material into a symmetric, customer ma
get_public_key	Returns the public key of an asymmetric CMK
import_key_material	Imports key material into an existing symmetric AWS KMS customer master key
list_aliases	Gets a list of aliases in the caller's AWS account and region
list_grants	Gets a list of all grants for the specified customer master key (CMK)
list_key_policies	Gets the names of the key policies that are attached to a customer master key (CM
list_keys	Gets a list of all customer master keys (CMKs) in the caller's AWS account and l
list_resource_tags	Returns a list of all tags for the specified customer master key (CMK)
list_retirable_grants	Returns a list of all grants for which the grant's RetiringPrincipal matches the on
put_key_policy	Attaches a key policy to the specified customer master key (CMK)
re_encrypt	Decrypts ciphertext and then reencrypts it entirely within AWS KMS
retire_grant	Retires a grant
revoke_grant	Revokes the specified grant for the specified customer master key (CMK)
schedule_key_deletion	Schedules the deletion of a customer master key (CMK)
sign	Creates a digital signature for a message or message digest by using the private k
tag_resource	Adds or edits tags for a customer master key (CMK)
untag_resource	Removes the specified tags from the specified customer master key (CMK)
update_alias	Associates an existing AWS KMS alias with a different customer master key (CM
update_custom_key_store	Changes the properties of a custom key store
update_key_description	Updates the description of a customer master key (CMK)
verify	Verifies a digital signature that was generated by the Sign operation

Examples

```
## Not run:
svc <- kms()
# The following example cancels deletion of the specified CMK.
svc$cancel_key_deletion(
  KeyId = "1234abcd-12ab-34cd-56ef-1234567890ab"
)
```

```
## End(Not run)
```

`macie` *Amazon Macie*

Description

Amazon Macie Classic

Amazon Macie Classic is a security service that uses machine learning to automatically discover, classify, and protect sensitive data in AWS. Macie Classic recognizes sensitive data such as personally identifiable information (PII) or intellectual property, and provides you with dashboards and alerts that give visibility into how this data is being accessed or moved. For more information, see the Amazon Macie Classic User Guide.

A new Amazon Macie is now available with significant design improvements and additional features, at a lower price and in most AWS Regions. We encourage you to explore and use the new and improved features, and benefit from the reduced cost. To learn about features and pricing for the new Amazon Macie, see Amazon Macie.

Usage

```
macie(config = list())
```

Arguments

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

Service syntax

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

Operations

associate_member_account	Associates a specified AWS account with Amazon Macie Classic as a member account
associate_s3_resources	Associates specified S3 resources with Amazon Macie Classic for monitoring and data classi
disassociate_member_account	Removes the specified member account from Amazon Macie Classic
disassociate_s3_resources	Removes specified S3 resources from being monitored by Amazon Macie Classic
list_member_accounts	Lists all Amazon Macie Classic member accounts for the current Amazon Macie Classic mas
list_s3_resources	Lists all the S3 resources associated with Amazon Macie Classic
update_s3_resources	Updates the classification types for the specified S3 resources

Examples

```
## Not run:
svc <- macie()
svc$associate_member_account(
  Foo = 123
)

## End(Not run)
```

---

ram *AWS Resource Access Manager*

Description

Use AWS Resource Access Manager to share AWS resources between AWS accounts. To share a resource, you create a resource share, associate the resource with the resource share, and specify the principals that can access the resources associated with the resource share. The following principals are supported: AWS accounts, organizational units (OU) from AWS Organizations, and organizations from AWS Organizations.

For more information, see the AWS Resource Access Manager User Guide.

Usage

```
ram(config = list())
```

Arguments

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

Service syntax

```
svc <- ram(
  config = list(
    credentials = list(
      creds = list(
```

```
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string"
  ),
  endpoint = "string",
  region = "string"
 )
)
```

Operations

accept_resource_share_invitation	Accepts an invitation to a resource share from another AWS account
associate_resource_share	Associates the specified resource share with the specified principals and reso
associate_resource_share_permission	Associates a permission with a resource share
create_resource_share	Creates a resource share
delete_resource_share	Deletes the specified resource share
disassociate_resource_share	Disassociates the specified principals or resources from the specified resourc
disassociate_resource_share_permission	Disassociates an AWS RAM permission from a resource share
enable_sharing_with_aws_organization	Enables resource sharing within your AWS Organization
get_permission	Gets the contents of an AWS RAM permission in JSON format
get_resource_policies	Gets the policies for the specified resources that you own and have shared
get_resource_share_associations	Gets the resources or principals for the resource shares that you own
get_resource_share_invitations	Gets the invitations for resource sharing that you've received
get_resource_shares	Gets the resource shares that you own or the resource shares that are shared v
list_pending_invitation_resources	Lists the resources in a resource share that is shared with you but that the inv
list_permissions	Lists the AWS RAM permissions
list_principals	Lists the principals that you have shared resources with or that have shared re
list_resources	Lists the resources that you added to a resource shares or the resources that a
list_resource_share_permissions	Lists the AWS RAM permissions that are associated with a resource share
list_resource_types	Lists the shareable resource types supported by AWS RAM
promote_resource_share_created_from_policy	Resource shares that were created by attaching a policy to a resource are visi
reject_resource_share_invitation	Rejects an invitation to a resource share from another AWS account
tag_resource	Adds the specified tags to the specified resource share that you own
untag_resource	Removes the specified tags from the specified resource share that you own
update_resource_share	Updates the specified resource share that you own

Examples

```
## Not run:
svc <- ram()
svc$accept_resource_share_invitation(
  Foo = 123
)

## End(Not run)
```

`secretsmanager` *AWS Secrets Manager*

Description

AWS Secrets Manager API Reference

AWS Secrets Manager provides a service to enable you to store, manage, and retrieve, secrets.

This guide provides descriptions of the Secrets Manager API. For more information about using this service, see the AWS Secrets Manager User Guide.

API Version

This version of the Secrets Manager API Reference documents the Secrets Manager API version 2017-10-17.

As an alternative to using the API, you can use one of the AWS SDKs, which consist of libraries and sample code for various programming languages and platforms such as Java, Ruby, .NET, iOS, and Android. The SDKs provide a convenient way to create programmatic access to AWS Secrets Manager. For example, the SDKs provide cryptographically signing requests, managing errors, and retrying requests automatically. For more information about the AWS SDKs, including downloading and installing them, see Tools for Amazon Web Services.

We recommend you use the AWS SDKs to make programmatic API calls to Secrets Manager. However, you also can use the Secrets Manager HTTP Query API to make direct calls to the Secrets Manager web service. To learn more about the Secrets Manager HTTP Query API, see Making Query Requests in the *AWS Secrets Manager User Guide*.

Secrets Manager API supports GET and POST requests for all actions, and doesn\'t require you to use GET for some actions and POST for others. However, GET requests are subject to the limitation size of a URL. Therefore, for operations that require larger sizes, use a POST request.

Support and Feedback for AWS Secrets Manager

We welcome your feedback. Send your comments to awssecretsmanager-feedback@amazon.com, or post your feedback and questions in the AWS Secrets Manager Discussion Forum. For more information about the AWS Discussion Forums, see Forums Help.

How examples are presented

The JSON that AWS Secrets Manager expects as your request parameters and the service returns as a response to HTTP query requests contain single, long strings without line breaks or white space formatting. The JSON shown in the examples displays the code formatted with both line breaks and white space to improve readability. When example input parameters can also cause long strings extending beyond the screen, you can insert line breaks to enhance readability. You should always submit the input as a single JSON text string.

Logging API Requests

AWS Secrets Manager supports AWS CloudTrail, a service that records AWS API calls for your AWS account and delivers log files to an Amazon S3 bucket. By using information that\'s collected by AWS CloudTrail, you can determine the requests successfully made to Secrets Manager, who

made the request, when it was made, and so on. For more about AWS Secrets Manager and support for AWS CloudTrail, see Logging AWS Secrets Manager Events with AWS CloudTrail in the *AWS Secrets Manager User Guide*. To learn more about CloudTrail, including enabling it and find your log files, see the AWS CloudTrail User Guide.

Usage

```
secretsmanager(config = list())
```

Arguments

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

Service syntax

```
svc <- secretsmanager(
  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_rotate_secret	Disables automatic scheduled rotation and cancels the rotation of a secret if currently in progre
create_secret	Creates a new secret
delete_resource_policy	Deletes the resource-based permission policy attached to the secret
delete_secret	Deletes an entire secret and all of its versions
describe_secret	Retrieves the details of a secret
get_random_password	Generates a random password of the specified complexity
get_resource_policy	Retrieves the JSON text of the resource-based policy document attached to the specified secret
get_secret_value	Retrieves the contents of the encrypted fields SecretString or SecretBinary from the specified v
list_secrets	Lists all of the secrets that are stored by Secrets Manager in the AWS account
list_secret_version_ids	Lists all of the versions attached to the specified secret
put_resource_policy	Attaches the contents of the specified resource-based permission policy to a secret
put_secret_value	Stores a new encrypted secret value in the specified secret
restore_secret	Cancels the scheduled deletion of a secret by removing the DeletedDate time stamp
rotate_secret	Configures and starts the asynchronous process of rotating this secret
tag_resource	Attaches one or more tags, each consisting of a key name and a value, to the specified secret
untag_resource	Removes one or more tags from the specified secret
update_secret	Modifies many of the details of the specified secret

update_secret_version_stage	Modifies the staging labels attached to a version of a secret
validate_resource_policy	Validates the JSON text of the resource-based policy document attached to the specified secret

Examples

```
## Not run:
svc <- secretsmanager()
# The following example shows how to cancel rotation for a secret. The
# operation sets the RotationEnabled field to false and cancels all
# scheduled rotations. To resume scheduled rotations, you must re-enable
# rotation by calling the rotate-secret operation.
svc$cancel_rotate_secret(
  SecretId = "MyTestDatabaseSecret"
)

## End(Not run)
```

`securityhub` *AWS SecurityHub*

Description

Security Hub provides you with a comprehensive view of the security state of your AWS environment and resources. It also provides you with the readiness status of your environment based on controls from supported security standards. Security Hub collects security data from AWS accounts, services, and integrated third-party products and helps you analyze security trends in your environment to identify the highest priority security issues. For more information about Security Hub, see the *AWS Security Hub User Guide* .

When you use operations in the Security Hub API, the requests are executed only in the AWS Region that is currently active or in the specific AWS Region that you specify in your request. Any configuration or settings change that results from the operation is applied only to that Region. To make the same change in other Regions, execute the same command for each Region to apply the change to.

For example, if your Region is set to `us-west-2`, when you use <a>CreateMembers to add a member account to Security Hub, the association of the member account with the master account is created only in the `us-west-2` Region. Security Hub must be enabled for the member account in the same Region that the invitation was sent from.

The following throttling limits apply to using Security Hub API operations.

- <a>GetFindings - `RateLimit` of 3 requests per second. `BurstLimit` of 6 requests per second.
- <a>UpdateFindings - `RateLimit` of 1 request per second. `BurstLimit` of 5 requests per second.
- All other operations - `RateLimit` of 10 requests per second. `BurstLimit` of 30 requests per second.

Usage

```
securityhub(config = list())
```

Arguments

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

Service syntax

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

Operations

accept_invitation	Accepts the invitation to be a member account and be monitored by the Security Hub
batch_disable_standards	Disables the standards specified by the provided StandardsSubscriptionArns
batch_enable_standards	Enables the standards specified by the provided StandardsArn
batch_import_findings	Imports security findings generated from an integrated third-party product into Securit
batch_update_findings	Used by Security Hub customers to update information about their investigation into a
create_action_target	Creates a custom action target in Security Hub
create_insight	Creates a custom insight in Security Hub
create_members	Creates a member association in Security Hub between the specified accounts and the
decline_invitations	Declines invitations to become a member account
delete_action_target	Deletes a custom action target from Security Hub
delete_insight	Deletes the insight specified by the InsightArn
delete_invitations	Deletes invitations received by the AWS account to become a member account
delete_members	Deletes the specified member accounts from Security Hub
describe_action_targets	Returns a list of the custom action targets in Security Hub in your account
describe_hub	Returns details about the Hub resource in your account, including the HubArn and the
describe_products	Returns information about the available products that you can subscribe to and integra
describe_standards	Returns a list of the available standards in Security Hub
describe_standards_controls	Returns a list of security standards controls
disable_import_findings_for_product	Disables the integration of the specified product with Security Hub
disable_security_hub	Disables Security Hub in your account only in the current Region
disassociate_from_master_account	Disassociates the current Security Hub member account from the associated master ac
disassociate_members	Disassociates the specified member accounts from the associated master account

enable_import_findings_for_product	Enables the integration of a partner product with Security Hub
enable_security_hub	Enables Security Hub for your account in the current Region or the Region you specif
get_enabled_standards	Returns a list of the standards that are currently enabled
get_findings	Returns a list of findings that match the specified criteria
get_insight_results	Lists the results of the Security Hub insight specified by the insight ARN
get_insights	Lists and describes insights for the specified insight ARNs
get_invitations_count	Returns the count of all Security Hub membership invitations that were sent to the cur
get_master_account	Provides the details for the Security Hub master account for the current member accou
get_members	Returns the details for the Security Hub member accounts for the specified account ID
invite_members	Invites other AWS accounts to become member accounts for the Security Hub master
list_enabled_products_for_import	Lists all findings-generating solutions (products) that you are subscribed to receive fin
list_invitations	Lists all Security Hub membership invitations that were sent to the current AWS accou
list_members	Lists details about all member accounts for the current Security Hub master account
list_tags_for_resource	Returns a list of tags associated with a resource
tag_resource	Adds one or more tags to a resource
untag_resource	Removes one or more tags from a resource
update_action_target	Updates the name and description of a custom action target in Security Hub
update_findings	UpdateFindings is deprecated
update_insight	Updates the Security Hub insight identified by the specified insight ARN
update_standards_control	Used to control whether an individual security standard control is enabled or disabled

Examples

```
## Not run:
svc <- securityhub()
svc$accept_invitation(
  Foo = 123
)

## End(Not run)
```

`shield` *AWS Shield*

Description

AWS Shield Advanced

This is the *AWS Shield Advanced API Reference*. This guide is for developers who need detailed information about the AWS Shield Advanced API actions, data types, and errors. For detailed information about AWS WAF and AWS Shield Advanced features and an overview of how to use the AWS WAF and AWS Shield Advanced APIs, see the AWS WAF and AWS Shield Developer Guide.

Usage

```
shield(config = list())
```

Arguments

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

Service syntax

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

Operations

associate_drt_log_bucket	Authorizes the DDoS Response Team (DRT) to access the specified Amazon S3 bu
associate_drt_role	Authorizes the DDoS Response Team (DRT), using the specified role, to access yo
associate_health_check	Adds health-based detection to the Shield Advanced protection for a resource
associate_proactive_engagement_details	Initializes proactive engagement and sets the list of contacts for the DDoS Respons
create_protection	Enables AWS Shield Advanced for a specific AWS resource
create_subscription	Activates AWS Shield Advanced for an account
delete_protection	Deletes an AWS Shield Advanced Protection
delete_subscription	Removes AWS Shield Advanced from an account
describe_attack	Describes the details of a DDoS attack
describe_drt_access	Returns the current role and list of Amazon S3 log buckets used by the DDoS Resp
describe_emergency_contact_settings	A list of email addresses and phone numbers that the DDoS Response Team (DRT)
describe_protection	Lists the details of a Protection object
describe_subscription	Provides details about the AWS Shield Advanced subscription for an account
disable_proactive_engagement	Removes authorization from the DDoS Response Team (DRT) to notify contacts al
disassociate_drt_log_bucket	Removes the DDoS Response Team's (DRT) access to the specified Amazon S3 bu
disassociate_drt_role	Removes the DDoS Response Team's (DRT) access to your AWS account
disassociate_health_check	Removes health-based detection from the Shield Advanced protection for a resourc
enable_proactive_engagement	Authorizes the DDoS Response Team (DRT) to use email and phone to notify cont
get_subscription_state	Returns the SubscriptionState, either Active or Inactive
list_attacks	Returns all ongoing DDoS attacks or all DDoS attacks during a specified time peri
list_protections	Lists all Protection objects for the account
update_emergency_contact_settings	Updates the details of the list of email addresses and phone numbers that the DDoS

update_subscription Updates the details of an existing subscription

Examples

```
## Not run:
svc <- shield()
svc$associate_drt_log_bucket(
  Foo = 123
)

## End(Not run)
```

---

sts *AWS Security Token Service*

Description

The AWS Security Token Service (STS) is a web service that enables you to request temporary, limited-privilege credentials for AWS Identity and Access Management (IAM) users or for users that you authenticate (federated users). This guide provides descriptions of the STS API. For more detailed information about using this service, go to Temporary Security Credentials.

For information about setting up signatures and authorization through the API, go to Signing AWS API Requests in the *AWS General Reference*. For general information about the Query API, go to Making Query Requests in *Using IAM*. For information about using security tokens with other AWS products, go to AWS Services That Work with IAM in the *IAM User Guide*.

If you\'re new to AWS and need additional technical information about a specific AWS product, you can find the product\'s technical documentation at http://aws.amazon.com/documentation/.

Endpoints

By default, AWS Security Token Service (STS) is available as a global service, and all AWS STS requests go to a single endpoint at https://sts.amazonaws.com. Global requests map to the US East (N. Virginia) region. AWS recommends using Regional AWS STS endpoints instead of the global endpoint to reduce latency, build in redundancy, and increase session token validity. For more information, see Managing AWS STS in an AWS Region in the *IAM User Guide*.

Most AWS Regions are enabled for operations in all AWS services by default. Those Regions are automatically activated for use with AWS STS. Some Regions, such as Asia Pacific (Hong Kong), must be manually enabled. To learn more about enabling and disabling AWS Regions, see Managing AWS Regions in the *AWS General Reference*. When you enable these AWS Regions, they are automatically activated for use with AWS STS. You cannot activate the STS endpoint for a Region that is disabled. Tokens that are valid in all AWS Regions are longer than tokens that are valid in Regions that are enabled by default. Changing this setting might affect existing systems where you temporarily store tokens. For more information, see Managing Global Endpoint Session Tokens in the *IAM User Guide*.

After you activate a Region for use with AWS STS, you can direct AWS STS API calls to that Region. AWS STS recommends that you provide both the Region and endpoint when you make calls to a Regional endpoint. You can provide the Region alone for manually enabled Regions, such as Asia Pacific (Hong Kong). In this case, the calls are directed to the STS Regional endpoint. However, if you provide the Region alone for Regions enabled by default, the calls are directed to the global endpoint of https://sts.amazonaws.com.

To view the list of AWS STS endpoints and whether they are active by default, see Writing Code to Use AWS STS Regions in the *IAM User Guide*.

Recording API requests

STS supports AWS CloudTrail, which is a service that records AWS calls for your AWS account and delivers log files to an Amazon S3 bucket. By using information collected by CloudTrail, you can determine what requests were successfully made to STS, who made the request, when it was made, and so on.

If you activate AWS STS endpoints in Regions other than the default global endpoint, then you must also turn on CloudTrail logging in those Regions. This is necessary to record any AWS STS API calls that are made in those Regions. For more information, see Turning On CloudTrail in Additional Regions in the *AWS CloudTrail User Guide*.

AWS Security Token Service (STS) is a global service with a single endpoint at https://sts.amazonaws.com. Calls to this endpoint are logged as calls to a global service. However, because this endpoint is physically located in the US East (N. Virginia) Region, your logs list `us-east-1` as the event Region. CloudTrail does not write these logs to the US East (Ohio) Region unless you choose to include global service logs in that Region. CloudTrail writes calls to all Regional endpoints to their respective Regions. For example, calls to sts.us-east-2.amazonaws.com are published to the US East (Ohio) Region and calls to sts.eu-central-1.amazonaws.com are published to the EU (Frankfurt) Region.

To learn more about CloudTrail, including how to turn it on and find your log files, see the AWS CloudTrail User Guide.

Usage

```
sts(config = list())
```

Arguments

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

Service syntax

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

```
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

assume_role	Returns a set of temporary security credentials that you can use to access AWS resources th
assume_role_with_saml	Returns a set of temporary security credentials for users who have been authenticated via a
assume_role_with_web_identity	Returns a set of temporary security credentials for users who have been authenticated in a r
decode_authorization_message	Decodes additional information about the authorization status of a request from an encoded
get_access_key_info	Returns the account identifier for the specified access key ID
get_caller_identity	Returns details about the IAM user or role whose credentials are used to call the operation
get_federation_token	Returns a set of temporary security credentials (consisting of an access key ID, a secret acc
get_session_token	Returns a set of temporary credentials for an AWS account or IAM user

Examples

```
## Not run:
svc <- sts()
#
svc$assume_role(
  ExternalId = "123ABC",
 Policy = "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":\"Stmt1\",\"Effect\":...",
  RoleArn = "arn:aws:iam::123456789012:role/demo",
  RoleSessionName = "testAssumeRoleSession",
  Tags = list(
    list(
      Key = "Project",
      Value = "Unicorn"
    ),
    list(
      Key = "Team",
      Value = "Automation"
    ),
    list(
      Key = "Cost-Center",
      Value = "12345"
    )
  ),
  TransitiveTagKeys = list(
    "Project",
    "Cost-Center"
  )
)

## End(Not run)
```

waf *AWS WAF*

Description

This is **AWS WAF Classic** documentation. For more information, see AWS WAF Classic in the developer guide.

For the latest version of AWS WAF, use the AWS WAFV2 API and see the AWS WAF Developer Guide. With the latest version, AWS WAF has a single set of endpoints for regional and global use.

This is the *AWS WAF Classic API Reference* for using AWS WAF Classic with Amazon CloudFront. The AWS WAF Classic actions and data types listed in the reference are available for protecting Amazon CloudFront distributions. You can use these actions and data types via the endpoint *waf.amazonaws.com*. This guide is for developers who need detailed information about the AWS WAF Classic API actions, data types, and errors. For detailed information about AWS WAF Classic features and an overview of how to use the AWS WAF Classic API, see the AWS WAF Classic in the developer guide.

Usage

```
waf(config = list())
```

Arguments

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

Service syntax

```
svc <- waf(
  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_byte_match_set	This is AWS WAF CLASSIC documentation
create_geo_match_set	This is AWS WAF CLASSIC documentation

create_ip_set	This is AWS WAF CLASSIC documentation
create_rate_based_rule	This is AWS WAF CLASSIC documentation
create_regex_match_set	This is AWS WAF CLASSIC documentation
create_regex_pattern_set	This is AWS WAF CLASSIC documentation
create_rule	This is AWS WAF CLASSIC documentation
create_rule_group	This is AWS WAF CLASSIC documentation
create_size_constraint_set	This is AWS WAF CLASSIC documentation
create_sql_injection_match_set	This is AWS WAF CLASSIC documentation
create_web_acl	This is AWS WAF CLASSIC documentation
create_web_acl_migration_stack	Creates an AWS CloudFormation WAFV2 template for the specified web ACL in the sp
create_xss_match_set	This is AWS WAF CLASSIC documentation
delete_byte_match_set	This is AWS WAF CLASSIC documentation
delete_geo_match_set	This is AWS WAF CLASSIC documentation
delete_ip_set	This is AWS WAF CLASSIC documentation
delete_logging_configuration	This is AWS WAF CLASSIC documentation
delete_permission_policy	This is AWS WAF CLASSIC documentation
delete_rate_based_rule	This is AWS WAF CLASSIC documentation
delete_regex_match_set	This is AWS WAF CLASSIC documentation
delete_regex_pattern_set	This is AWS WAF CLASSIC documentation
delete_rule	This is AWS WAF CLASSIC documentation
delete_rule_group	This is AWS WAF CLASSIC documentation
delete_size_constraint_set	This is AWS WAF CLASSIC documentation
delete_sql_injection_match_set	This is AWS WAF CLASSIC documentation
delete_web_acl	This is AWS WAF CLASSIC documentation
delete_xss_match_set	This is AWS WAF CLASSIC documentation
get_byte_match_set	This is AWS WAF CLASSIC documentation
get_change_token	This is AWS WAF CLASSIC documentation
get_change_token_status	This is AWS WAF CLASSIC documentation
get_geo_match_set	This is AWS WAF CLASSIC documentation
get_ip_set	This is AWS WAF CLASSIC documentation
get_logging_configuration	This is AWS WAF CLASSIC documentation
get_permission_policy	This is AWS WAF CLASSIC documentation
get_rate_based_rule	This is AWS WAF CLASSIC documentation
get_rate_based_rule_managed_keys	This is AWS WAF CLASSIC documentation
get_regex_match_set	This is AWS WAF CLASSIC documentation
get_regex_pattern_set	This is AWS WAF CLASSIC documentation
get_rule	This is AWS WAF CLASSIC documentation
get_rule_group	This is AWS WAF CLASSIC documentation
get_sampled_requests	This is AWS WAF CLASSIC documentation
get_size_constraint_set	This is AWS WAF CLASSIC documentation
get_sql_injection_match_set	This is AWS WAF CLASSIC documentation
get_web_acl	This is AWS WAF CLASSIC documentation
get_xss_match_set	This is AWS WAF CLASSIC documentation
list_activated_rules_in_rule_group	This is AWS WAF CLASSIC documentation
list_byte_match_sets	This is AWS WAF CLASSIC documentation
list_geo_match_sets	This is AWS WAF CLASSIC documentation
list_ip_sets	This is AWS WAF CLASSIC documentation
list_logging_configurations	This is AWS WAF CLASSIC documentation

list_rate_based_rules	This is AWS WAF CLASSIC documentation
list_regex_match_sets	This is AWS WAF CLASSIC documentation
list_regex_pattern_sets	This is AWS WAF CLASSIC documentation
list_rule_groups	This is AWS WAF CLASSIC documentation
list_rules	This is AWS WAF CLASSIC documentation
list_size_constraint_sets	This is AWS WAF CLASSIC documentation
list_sql_injection_match_sets	This is AWS WAF CLASSIC documentation
list_subscribed_rule_groups	This is AWS WAF CLASSIC documentation
list_tags_for_resource	This is AWS WAF CLASSIC documentation
list_web_ac_ls	This is AWS WAF CLASSIC documentation
list_xss_match_sets	This is AWS WAF CLASSIC documentation
put_logging_configuration	This is AWS WAF CLASSIC documentation
put_permission_policy	This is AWS WAF CLASSIC documentation
tag_resource	This is AWS WAF CLASSIC documentation
untag_resource	This is AWS WAF CLASSIC documentation
update_byte_match_set	This is AWS WAF CLASSIC documentation
update_geo_match_set	This is AWS WAF CLASSIC documentation
update_ip_set	This is AWS WAF CLASSIC documentation
update_rate_based_rule	This is AWS WAF CLASSIC documentation
update_regex_match_set	This is AWS WAF CLASSIC documentation
update_regex_pattern_set	This is AWS WAF CLASSIC documentation
update_rule	This is AWS WAF CLASSIC documentation
update_rule_group	This is AWS WAF CLASSIC documentation
update_size_constraint_set	This is AWS WAF CLASSIC documentation
update_sql_injection_match_set	This is AWS WAF CLASSIC documentation
update_web_acl	This is AWS WAF CLASSIC documentation
update_xss_match_set	This is AWS WAF CLASSIC documentation

Examples

```
## Not run:
svc <- waf()
# The following example creates an IP match set named MyIPSetFriendlyName.
svc$create_ip_set(
  ChangeToken = "abcd12f2-46da-4fdb-b8d5-fbd4c466928f",
  Name = "MyIPSetFriendlyName"
)

## End(Not run)
```

`wafregional` *AWS WAF Regional*

Description

This is **AWS WAF Classic Regional** documentation. For more information, see AWS WAF Classic in the developer guide.

For the latest version of AWS WAF, use the AWS WAFV2 API and see the AWS WAF Developer Guide. With the latest version, AWS WAF has a single set of endpoints for regional and global use.

This is the *AWS WAF Regional Classic API Reference* for using AWS WAF Classic with the AWS resources, Elastic Load Balancing (ELB) Application Load Balancers and API Gateway APIs. The AWS WAF Classic actions and data types listed in the reference are available for protecting Elastic Load Balancing (ELB) Application Load Balancers and API Gateway APIs. You can use these actions and data types by means of the endpoints listed in AWS Regions and Endpoints. This guide is for developers who need detailed information about the AWS WAF Classic API actions, data types, and errors. For detailed information about AWS WAF Classic features and an overview of how to use the AWS WAF Classic API, see the AWS WAF Classic in the developer guide.

Usage

```
wafregional(config = list())
```

Arguments

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

Service syntax

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

Operations

associate_web_acl	This is AWS WAF CLASSIC REGIONAL documentation
create_byte_match_set	This is AWS WAF CLASSIC documentation
create_geo_match_set	This is AWS WAF CLASSIC documentation
create_ip_set	This is AWS WAF CLASSIC documentation
create_rate_based_rule	This is AWS WAF CLASSIC documentation
create_regex_match_set	This is AWS WAF CLASSIC documentation
create_regex_pattern_set	This is AWS WAF CLASSIC documentation

create_rule	This is AWS WAF CLASSIC documentation
create_rule_group	This is AWS WAF CLASSIC documentation
create_size_constraint_set	This is AWS WAF CLASSIC documentation
create_sql_injection_match_set	This is AWS WAF CLASSIC documentation
create_web_acl	This is AWS WAF CLASSIC documentation
create_web_acl_migration_stack	Creates an AWS CloudFormation WAFV2 template for the specified web ACL in the sp
create_xss_match_set	This is AWS WAF CLASSIC documentation
delete_byte_match_set	This is AWS WAF CLASSIC documentation
delete_geo_match_set	This is AWS WAF CLASSIC documentation
delete_ip_set	This is AWS WAF CLASSIC documentation
delete_logging_configuration	This is AWS WAF CLASSIC documentation
delete_permission_policy	This is AWS WAF CLASSIC documentation
delete_rate_based_rule	This is AWS WAF CLASSIC documentation
delete_regex_match_set	This is AWS WAF CLASSIC documentation
delete_regex_pattern_set	This is AWS WAF CLASSIC documentation
delete_rule	This is AWS WAF CLASSIC documentation
delete_rule_group	This is AWS WAF CLASSIC documentation
delete_size_constraint_set	This is AWS WAF CLASSIC documentation
delete_sql_injection_match_set	This is AWS WAF CLASSIC documentation
delete_web_acl	This is AWS WAF CLASSIC documentation
delete_xss_match_set	This is AWS WAF CLASSIC documentation
disassociate_web_acl	This is AWS WAF CLASSIC REGIONAL documentation
get_byte_match_set	This is AWS WAF CLASSIC documentation
get_change_token	This is AWS WAF CLASSIC documentation
get_change_token_status	This is AWS WAF CLASSIC documentation
get_geo_match_set	This is AWS WAF CLASSIC documentation
get_ip_set	This is AWS WAF CLASSIC documentation
get_logging_configuration	This is AWS WAF CLASSIC documentation
get_permission_policy	This is AWS WAF CLASSIC documentation
get_rate_based_rule	This is AWS WAF CLASSIC documentation
get_rate_based_rule_managed_keys	This is AWS WAF CLASSIC documentation
get_regex_match_set	This is AWS WAF CLASSIC documentation
get_regex_pattern_set	This is AWS WAF CLASSIC documentation
get_rule	This is AWS WAF CLASSIC documentation
get_rule_group	This is AWS WAF CLASSIC documentation
get_sampled_requests	This is AWS WAF CLASSIC documentation
get_size_constraint_set	This is AWS WAF CLASSIC documentation
get_sql_injection_match_set	This is AWS WAF CLASSIC documentation
get_web_acl	This is AWS WAF CLASSIC documentation
get_web_acl_for_resource	This is AWS WAF CLASSIC REGIONAL documentation
get_xss_match_set	This is AWS WAF CLASSIC documentation
list_activated_rules_in_rule_group	This is AWS WAF CLASSIC documentation
list_byte_match_sets	This is AWS WAF CLASSIC documentation
list_geo_match_sets	This is AWS WAF CLASSIC documentation
list_ip_sets	This is AWS WAF CLASSIC documentation
list_logging_configurations	This is AWS WAF CLASSIC documentation
list_rate_based_rules	This is AWS WAF CLASSIC documentation
list_regex_match_sets	This is AWS WAF CLASSIC documentation

list_regex_pattern_sets	This is AWS WAF CLASSIC documentation
list_resources_for_web_acl	This is AWS WAF CLASSIC REGIONAL documentation
list_rule_groups	This is AWS WAF CLASSIC documentation
list_rules	This is AWS WAF CLASSIC documentation
list_size_constraint_sets	This is AWS WAF CLASSIC documentation
list_sql_injection_match_sets	This is AWS WAF CLASSIC documentation
list_subscribed_rule_groups	This is AWS WAF CLASSIC documentation
list_tags_for_resource	This is AWS WAF CLASSIC documentation
list_web_ac_ls	This is AWS WAF CLASSIC documentation
list_xss_match_sets	This is AWS WAF CLASSIC documentation
put_logging_configuration	This is AWS WAF CLASSIC documentation
put_permission_policy	This is AWS WAF CLASSIC documentation
tag_resource	This is AWS WAF CLASSIC documentation
untag_resource	This is AWS WAF CLASSIC documentation
update_byte_match_set	This is AWS WAF CLASSIC documentation
update_geo_match_set	This is AWS WAF CLASSIC documentation
update_ip_set	This is AWS WAF CLASSIC documentation
update_rate_based_rule	This is AWS WAF CLASSIC documentation
update_regex_match_set	This is AWS WAF CLASSIC documentation
update_regex_pattern_set	This is AWS WAF CLASSIC documentation
update_rule	This is AWS WAF CLASSIC documentation
update_rule_group	This is AWS WAF CLASSIC documentation
update_size_constraint_set	This is AWS WAF CLASSIC documentation
update_sql_injection_match_set	This is AWS WAF CLASSIC documentation
update_web_acl	This is AWS WAF CLASSIC documentation
update_xss_match_set	This is AWS WAF CLASSIC documentation

Examples

```
## Not run:
svc <- wafregional()
# The following example creates an IP match set named MyIPSetFriendlyName.
svc$create_ip_set(
  ChangeToken = "abcd12f2-46da-4fdb-b8d5-fbd4c466928f",
  Name = "MyIPSetFriendlyName"
)

## End(Not run)
```

Index