Package 'paws.compute'

July 23, 2025

Title 'Amazon Web Services' Compute Services

Version 0.9.0

Description Interface to 'Amazon Web Services' compute services, including 'Elastic Compute Cloud' ('EC2'), 'Lambda' functions-as-a-service, containers, batch processing, and more <https://aws.amazon.com/>.

License Apache License (>= 2.0)

URL https://github.com/paws-r/paws, https://paws-r.r-universe.dev/paws.compute

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

Imports paws.common (>= 0.8.0)

Suggests testthat

Encoding UTF-8

RoxygenNote 7.3.2

Collate 'apprunner_service.R' 'apprunner_interfaces.R' 'apprunner_operations.R' 'batch_service.R' 'batch_interfaces.R' 'batch_operations.R' 'braket_service.R' 'braket_interfaces.R' 'braket_operations.R' 'computeoptimizer_service.R' 'computeoptimizer_interfaces.R' 'computeoptimizer_operations.R' 'ec2_service.R' 'ec2_interfaces.R' 'ec2_operations.R' 'ec2instanceconnect_service.R' 'ec2instanceconnect_interfaces.R' 'ec2instanceconnect_operations.R' 'ecr_service.R' 'ecr_interfaces.R' 'ecr_operations.R' 'ecrpublic_service.R' 'ecrpublic_interfaces.R' 'ecrpublic_operations.R' 'ecs_service.R' 'ecs_interfaces.R' 'ecs_operations.R' 'eks_service.R' 'eks_interfaces.R' 'eks_operations.R' 'elasticbeanstalk_service.R' 'elasticbeanstalk_interfaces.R' 'elasticbeanstalk_operations.R' 'emrcontainers_service.R' 'emrcontainers_interfaces.R' 'emrcontainers_operations.R' 'emrserverless_service.R' 'emrserverless_interfaces.R' 'emrserverless_operations.R' 'imagebuilder_service.R'

'imagebuilder_interfaces.R' 'imagebuilder_operations.R'
'lambda_service.R' 'lambda_interfaces.R' 'lambda_operations.R'
'lightsail_service.R' 'lightsail_interfaces.R'
'lightsail_operations.R' 'proton_service.R'
'proton_interfaces.R' 'proton_operations.R'
'reexports_paws.common.R'
'serverlessapplicationrepository_service.R'
'serverlessapplicationrepository_interfaces.R'
'serverlessapplicationrepository_operations.R'

NeedsCompilation no

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

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

Repository CRAN

Date/Publication 2025-03-14 14:00:02 UTC

Contents

`apprunner` *AWS App Runner*

Description

App Runner

App Runner is an application service that provides a fast, simple, and cost-effective way to go directly from an existing container image or source code to a running service in the Amazon Web Services Cloud in seconds. You don't need to learn new technologies, decide which compute service to use, or understand how to provision and configure Amazon Web Services resources.

App Runner connects directly to your container registry or source code repository. It provides an automatic delivery pipeline with fully managed operations, high performance, scalability, and security.

For more information about App Runner, see the App Runner Developer Guide. For release information, see the App Runner Release Notes.

To install the Software Development Kits (SDKs), Integrated Development Environment (IDE) Toolkits, and command line tools that you can use to access the API, see Tools for Amazon Web Services.

Endpoints

For a list of Region-specific endpoints that App Runner supports, see App Runner endpoints and quotas in the *Amazon Web Services General Reference*.

Usage

```
apprunner(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

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

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

- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-e html

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
  )
```

Operations

associate_custom_domain	Associate your own domain name with the App Runner subdomain URL of yo
create_auto_scaling_configuration	Create an App Runner automatic scaling configuration resource
create_connection	Create an App Runner connection resource
create_observability_configuration	Create an App Runner observability configuration resource
create_service	Create an App Runner service
create_vpc_connector	Create an App Runner VPC connector resource
create_vpc_ingress_connection	Create an App Runner VPC Ingress Connection resource
delete_auto_scaling_configuration	Delete an App Runner automatic scaling configuration resource
delete_connection	Delete an App Runner connection
delete_observability_configuration	Delete an App Runner observability configuration resource
delete_service	Delete an App Runner service
delete_vpc_connector	Delete an App Runner VPC connector resource
delete_vpc_ingress_connection	Delete an App Runner VPC Ingress Connection resource that's associated with
describe_auto_scaling_configuration	Return a full description of an App Runner automatic scaling configuration res
describe_custom_domains	Return a description of custom domain names that are associated with an App
describe_observability_configuration	Return a full description of an App Runner observability configuration resourc
describe_service	Return a full description of an App Runner service
describe_vpc_connector	Return a description of an App Runner VPC connector resource
describe_vpc_ingress_connection	Return a full description of an App Runner VPC Ingress Connection resource
disassociate_custom_domain	Disassociate a custom domain name from an App Runner service
list_auto_scaling_configurations	Returns a list of active App Runner automatic scaling configurations in your A
list_connections	Returns a list of App Runner connections that are associated with your Amazo
list_observability_configurations	Returns a list of active App Runner observability configurations in your Amaz
list_operations	Return a list of operations that occurred on an App Runner service
list_services	Returns a list of running App Runner services in your Amazon Web Services a
list_services_for_auto_scaling_configuration	Returns a list of the associated App Runner services using an auto scaling conf
list_tags_for_resource	List tags that are associated with for an App Runner resource
list_vpc_connectors	Returns a list of App Runner VPC connectors in your Amazon Web Services a
list_vpc_ingress_connections	Return a list of App Runner VPC Ingress Connections in your Amazon Web S
pause_service	Pause an active App Runner service
resume_service	Resume an active App Runner service
start_deployment	Initiate a manual deployment of the latest commit in a source code repository
tag_resource	Add tags to, or update the tag values of, an App Runner resource

untag_resource	Remove tags from an App Runner resource
update_default_auto_scaling_configuration	Update an auto scaling configuration to be the default
update_service	Update an App Runner service
update_vpc_ingress_connection	Update an existing App Runner VPC Ingress Connection resource

Examples

```
## Not run:
svc <- apprunner()
svc$associate_custom_domain(
  Foo = 123
)

## End(Not run)
```

`batch` *AWS Batch*

Description

Batch

Using Batch, you can run batch computing workloads on the Amazon Web Services Cloud. Batch computing is a common means for developers, scientists, and engineers to access large amounts of compute resources. Batch uses the advantages of the batch computing to remove the undifferentiated heavy lifting of configuring and managing required infrastructure. At the same time, it also adopts a familiar batch computing software approach. You can use Batch to efficiently provision resources, and work toward eliminating capacity constraints, reducing your overall compute costs, and delivering results more quickly.

As a fully managed service, Batch can run batch computing workloads of any scale. Batch automatically provisions compute resources and optimizes workload distribution based on the quantity and scale of your specific workloads. With Batch, there's no need to install or manage batch computing software. This means that you can focus on analyzing results and solving your specific problems instead.

Usage

```
batch(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

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

- **credentials**:
  - **creds**:

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

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

cancel_job	Cancels a job in an Batch job queue
create_compute_environment	Creates an Batch compute environment
create_job_queue	Creates an Batch job queue
create_scheduling_policy	Creates an Batch scheduling policy
delete_compute_environment	Deletes an Batch compute environment
delete_job_queue	Deletes the specified job queue
delete_scheduling_policy	Deletes the specified scheduling policy
deregister_job_definition	Deregisters an Batch job definition
describe_compute_environments	Describes one or more of your compute environments
describe_job_definitions	Describes a list of job definitions
describe_job_queues	Describes one or more of your job queues
describe_jobs	Describes a list of Batch jobs
describe_scheduling_policies	Describes one or more of your scheduling policies
get_job_queue_snapshot	Provides a list of the first 100 RUNNABLE jobs associated to a single job queue
list_jobs	Returns a list of Batch jobs
list_scheduling_policies	Returns a list of Batch scheduling policies
list_tags_for_resource	Lists the tags for an Batch resource
register_job_definition	Registers an Batch job definition
submit_job	Submits an Batch job from a job definition
tag_resource	Associates the specified tags to a resource with the specified resourceArn
terminate_job	Terminates a job in a job queue
untag_resource	Deletes specified tags from an Batch resource
update_compute_environment	Updates an Batch compute environment
update_job_queue	Updates a job queue
update_scheduling_policy	Updates a scheduling policy

Examples

```
## Not run:
svc <- batch()
# This example cancels a job with the specified job ID.
svc$cancel_job(
  jobId = "1d828f65-7a4d-42e8-996d-3b900ed59dc4",
  reason = "Cancelling job."
)

## End(Not run)
```

`braket` *Braket*

Description

The Amazon Braket API Reference provides information about the operations and structures supported in Amazon Braket.

Additional Resources:

- Amazon Braket Developer Guide

Usage

```
braket(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

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

- **credentials**:
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-e html

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

cancel_job	Cancels an Amazon Braket job
cancel_quantum_task	Cancels the specified task
create_job	Creates an Amazon Braket job
create_quantum_task	Creates a quantum task
get_device	Retrieves the devices available in Amazon Braket
get_job	Retrieves the specified Amazon Braket job
get_quantum_task	Retrieves the specified quantum task
list_tags_for_resource	Shows the tags associated with this resource
search_devices	Searches for devices using the specified filters
search_jobs	Searches for Amazon Braket jobs that match the specified filter values
search_quantum_tasks	Searches for tasks that match the specified filter values
tag_resource	Add a tag to the specified resource
untag_resource	Remove tags from a resource

Examples

```
## Not run:
svc <- braket()
svc$cancel_job(
  Foo = 123
)

## End(Not run)
```

---

computeoptimizer *AWS Compute Optimizer*

Description

Compute Optimizer is a service that analyzes the configuration and utilization metrics of your Amazon Web Services compute resources, such as Amazon EC2 instances, Amazon EC2 Auto Scaling groups, Lambda functions, Amazon EBS volumes, and Amazon ECS services on Fargate. It reports whether your resources are optimal, and generates optimization recommendations to reduce the cost

and improve the performance of your workloads. Compute Optimizer also provides recent utilization metric data, in addition to projected utilization metric data for the recommendations, which you can use to evaluate which recommendation provides the best price-performance trade-off. The analysis of your usage patterns can help you decide when to move or resize your running resources, and still meet your performance and capacity requirements. For more information about Compute Optimizer, including the required permissions to use the service, see the Compute Optimizer User Guide.

Usage

```
computeoptimizer(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

Operations

delete_recommendation_preferences	Deletes a recommendation preference, such as enhanced infrastructu
describe_recommendation_export_jobs	Describes recommendation export jobs created in the last seven days
export_auto_scaling_group_recommendations	Exports optimization recommendations for Auto Scaling groups

export_ebs_volume_recommendations	Exports optimization recommendations for Amazon EBS volumes
export_ec2_instance_recommendations	Exports optimization recommendations for Amazon EC2 instances
export_ecs_service_recommendations	Exports optimization recommendations for Amazon ECS services on
export_idle_recommendations	Export optimization recommendations for your idle resources
export_lambda_function_recommendations	Exports optimization recommendations for Lambda functions
export_license_recommendations	Export optimization recommendations for your licenses
export_rds_database_recommendations	Export optimization recommendations for your Amazon Relational D
get_auto_scaling_group_recommendations	Returns Auto Scaling group recommendations
get_ebs_volume_recommendations	Returns Amazon Elastic Block Store (Amazon EBS) volume recomn
get_ec2_instance_recommendations	Returns Amazon EC2 instance recommendations
get_ec2_recommendation_projected_metrics	Returns the projected utilization metrics of Amazon EC2 instance rec
get_ecs_service_recommendation_projected_metrics	Returns the projected metrics of Amazon ECS service recommendati
get_ecs_service_recommendations	Returns Amazon ECS service recommendations
get_effective_recommendation_preferences	Returns the recommendation preferences that are in effect for a given
get_enrollment_status	Returns the enrollment (opt in) status of an account to the Compute C
get_enrollment_statuses_for_organization	Returns the Compute Optimizer enrollment (opt-in) status of organiz
get_idle_recommendations	Returns idle resource recommendations
get_lambda_function_recommendations	Returns Lambda function recommendations
get_license_recommendations	Returns license recommendations for Amazon EC2 instances that run
get_rds_database_recommendation_projected_metrics	Returns the projected metrics of Amazon RDS recommendations
get_rds_database_recommendations	Returns Amazon RDS recommendations
get_recommendation_preferences	Returns existing recommendation preferences, such as enhanced infr
get_recommendation_summaries	Returns the optimization findings for an account
put_recommendation_preferences	Creates a new recommendation preference or updates an existing rec
update_enrollment_status	Updates the enrollment (opt in and opt out) status of an account to th

Examples

```
## Not run:
svc <- computeoptimizer()
svc$delete_recommendation_preferences(
  Foo = 123
)

## End(Not run)
```

`ec2` *Amazon Elastic Compute Cloud*

Description

You can access the features of Amazon Elastic Compute Cloud (Amazon EC2) programmatically. For more information, see the Amazon EC2 Developer Guide.

Usage

```
ec2(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

config	Optional configuration of credentials, endpoint, and/or region. • **credentials**: – **creds**: ∗ **access_key_id**: AWS access key ID ∗ **secret_access_key**: AWS secret access key ∗ **session_token**: AWS temporary session token – **profile**: The name of a profile to use. If not given, then the default profile is used. – **anonymous**: Set anonymous credentials. • **endpoint**: The complete URL to use for the constructed client. • **region**: The AWS Region used in instantiating the client. • **close_connection**: Immediately close all HTTP connections. • **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds. • **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`. • **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-e html
credentials	Optional credentials shorthand for the config parameter • **creds**: – **access_key_id**: AWS access key ID – **secret_access_key**: AWS secret access key – **session_token**: AWS temporary session token • **profile**: The name of a profile to use. If not given, then the default profile is used. • **anonymous**: Set anonymous credentials.
endpoint	Optional shorthand for complete URL to use for the constructed client.
region	Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

```
svc <- ec2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

accept_address_transfer	Accepts an Elastic IP address transfer
accept_capacity_reservation_billing_ownership	Accepts a request to assign billing of the availabl
accept_reserved_instances_exchange_quote	Accepts the Convertible Reserved Instance excha
accept_transit_gateway_multicast_domain_associations	Accepts a request to associate subnets with a tran
accept_transit_gateway_peering_attachment	Accepts a transit gateway peering attachment req
accept_transit_gateway_vpc_attachment	Accepts a request to attach a VPC to a transit gat
accept_vpc_endpoint_connections	Accepts connection requests to your VPC endpoi
accept_vpc_peering_connection	Accept a VPC peering connection request
advertise_byoip_cidr	Advertises an IPv4 or IPv6 address range that is
allocate_address	Allocates an Elastic IP address to your Amazon V
allocate_hosts	Allocates a Dedicated Host to your account
allocate_ipam_pool_cidr	Allocate a CIDR from an IPAM pool
apply_security_groups_to_client_vpn_target_network	Applies a security group to the association betwe

assign_ipv_6_addresses	Assigns the specified IPv6 addresses to the speci
assign_private_ip_addresses	Assigns the specified secondary private IP addres
assign_private_nat_gateway_address	Assigns private IPv4 addresses to a private NAT
associate_address	Associates an Elastic IP address, or carrier IP ad
associate_capacity_reservation_billing_owner	Initiates a request to assign billing of the unused
associate_client_vpn_target_network	Associates a target network with a Client VPN e
associate_dhcp_options	Associates a set of DHCP options (that you've pr
associate_enclave_certificate_iam_role	Associates an Identity and Access Management
associate_iam_instance_profile	Associates an IAM instance profile with a runnin
associate_instance_event_window	Associates one or more targets with an event win
associate_ipam_byoasn	Associates your Autonomous System Number (A
associate_ipam_resource_discovery	Associates an IPAM resource discovery with an
associate_nat_gateway_address	Associates Elastic IP addresses (EIPs) and privat
associate_route_table	Associates a subnet in your VPC or an internet g
associate_security_group_vpc	Associates a security group with another VPC in
associate_subnet_cidr_block	Associates a CIDR block with your subnet
associate_transit_gateway_multicast_domain	Associates the specified subnets and transit gatev
associate_transit_gateway_policy_table	Associates the specified transit gateway attachme
associate_transit_gateway_route_table	Associates the specified attachment with the spec
associate_trunk_interface	Associates a branch network interface with a tru
associate_vpc_cidr_block	Associates a CIDR block with your VPC
attach_classic_link_vpc	This action is deprecated
attach_internet_gateway	Attaches an internet gateway or a virtual private
attach_network_interface	Attaches a network interface to an instance
attach_verified_access_trust_provider	Attaches the specified Amazon Web Services Ve
attach_volume	Attaches an EBS volume to a running or stopped
attach_vpn_gateway	Attaches an available virtual private gateway to a
authorize_client_vpn_ingress	Adds an ingress authorization rule to a Client VP
authorize_security_group_egress	Adds the specified outbound (egress) rules to a s
authorize_security_group_ingress	Adds the specified inbound (ingress) rules to a se
bundle_instance	Bundles an Amazon instance store-backed Windo
cancel_bundle_task	Cancels a bundling operation for an instance stor
cancel_capacity_reservation	Cancels the specified Capacity Reservation, relea
cancel_capacity_reservation_fleets	Cancels one or more Capacity Reservation Fleets
cancel_conversion_task	Cancels an active conversion task
cancel_declarative_policies_report	Cancels the generation of an account status repor
cancel_export_task	Cancels an active export task
cancel_image_launch_permission	Removes your Amazon Web Services account fr
cancel_import_task	Cancels an in-process import virtual machine or
cancel_reserved_instances_listing	Cancels the specified Reserved Instance listing in
cancel_spot_fleet_requests	Cancels the specified Spot Fleet requests
cancel_spot_instance_requests	Cancels one or more Spot Instance requests
confirm_product_instance	Determines whether a product code is associated
copy_fpga_image	Copies the specified Amazon FPGA Image (AFI
copy_image	Initiates an AMI copy operation
copy_snapshot	Copies a point-in-time snapshot of an EBS volun
create_capacity_reservation	Creates a new Capacity Reservation with the spe
create_capacity_reservation_by_splitting	Create a new Capacity Reservation by splitting th

create_capacity_reservation_fleet	Creates a Capacity Reservation Fleet
create_carrier_gateway	Creates a carrier gateway
create_client_vpn_endpoint	Creates a Client VPN endpoint
create_client_vpn_route	Adds a route to a network to a Client VPN endpo
create_coip_cidr	Creates a range of customer-owned IP addresses
create_coip_pool	Creates a pool of customer-owned IP (CoIP) add
create_customer_gateway	Provides information to Amazon Web Services a
create_default_subnet	Creates a default subnet with a size /20 IPv4 CID
create_default_vpc	Creates a default VPC with a size /16 IPv4 CIDR
create_dhcp_options	Creates a custom set of DHCP options
create_egress_only_internet_gateway	[IPv6 only] Creates an egress-only internet gatev
create_fleet	Creates an EC2 Fleet that contains the configurat
create_flow_logs	Creates one or more flow logs to capture informa
create_fpga_image	Creates an Amazon FPGA Image (AFI) from the
create_image	Creates an Amazon EBS-backed AMI from an A
create_instance_connect_endpoint	Creates an EC2 Instance Connect Endpoint
create_instance_event_window	Creates an event window in which scheduled eve
create_instance_export_task	Exports a running or stopped instance to an Ama
create_internet_gateway	Creates an internet gateway for use with a VPC
create_ipam	Create an IPAM
create_ipam_external_resource_verification_token	Create a verification token
create_ipam_pool	Create an IP address pool for Amazon VPC IP A
create_ipam_resource_discovery	Creates an IPAM resource discovery
create_ipam_scope	Create an IPAM scope
create_key_pair	Creates an ED25519 or 2048-bit RSA key pair w
create_launch_template	Creates a launch template
create_launch_template_version	Creates a new version of a launch template
create_local_gateway_route	Creates a static route for the specified local gatev
create_local_gateway_route_table	Creates a local gateway route table
create_local_gateway_route_table_virtual_interface_group_association	Creates a local gateway route table virtual interfa
create_local_gateway_route_table_vpc_association	Associates the specified VPC with the specified l
create_managed_prefix_list	Creates a managed prefix list
create_nat_gateway	Creates a NAT gateway in the specified subnet
create_network_acl	Creates a network ACL in a VPC
create_network_acl_entry	Creates an entry (a rule) in a network ACL with t
create_network_insights_access_scope	Creates a Network Access Scope
create_network_insights_path	Creates a path to analyze for reachability
create_network_interface	Creates a network interface in the specified subn
create_network_interface_permission	Grants an Amazon Web Services-authorized acc
create_placement_group	Creates a placement group in which to launch ins
create_public_ipv_4_pool	Creates a public IPv4 address pool
create_replace_root_volume_task	Replaces the EBS-backed root volume for a runn
create_reserved_instances_listing	Creates a listing for Amazon EC2 Standard Rese
create_restore_image_task	Starts a task that restores an AMI from an Amaz
create_route	Creates a route in a route table within a VPC
create_route_table	Creates a route table for the specified VPC
create_security_group	Creates a security group
create_snapshot	Creates a snapshot of an EBS volume and stores

create_snapshots	Creates crash-consistent snapshots of multiple El
create_spot_datafeed_subscription	Creates a data feed for Spot Instances, enabling y
create_store_image_task	Stores an AMI as a single object in an Amazon S
create_subnet	Creates a subnet in the specified VPC
create_subnet_cidr_reservation	Creates a subnet CIDR reservation
create_tags	Adds or overwrites only the specified tags for the
create_traffic_mirror_filter	Creates a Traffic Mirror filter
create_traffic_mirror_filter_rule	Creates a Traffic Mirror filter rule
create_traffic_mirror_session	Creates a Traffic Mirror session
create_traffic_mirror_target	Creates a target for your Traffic Mirror session
create_transit_gateway	Creates a transit gateway
create_transit_gateway_connect	Creates a Connect attachment from a specified tr
create_transit_gateway_connect_peer	Creates a Connect peer for a specified transit gat
create_transit_gateway_multicast_domain	Creates a multicast domain using the specified tr
create_transit_gateway_peering_attachment	Requests a transit gateway peering attachment be
create_transit_gateway_policy_table	Creates a transit gateway policy table
create_transit_gateway_prefix_list_reference	Creates a reference (route) to a prefix list in a spe
create_transit_gateway_route	Creates a static route for the specified transit gate
create_transit_gateway_route_table	Creates a route table for the specified transit gate
create_transit_gateway_route_table_announcement	Advertises a new transit gateway route table
create_transit_gateway_vpc_attachment	Attaches the specified VPC to the specified trans
create_verified_access_endpoint	An Amazon Web Services Verified Access endpo
create_verified_access_group	An Amazon Web Services Verified Access group
create_verified_access_instance	An Amazon Web Services Verified Access instan
create_verified_access_trust_provider	A trust provider is a third-party entity that create
create_volume	Creates an EBS volume that can be attached to a
create_vpc	Creates a VPC with the specified CIDR blocks
create_vpc_block_public_access_exclusion	Create a VPC Block Public Access (BPA) exclus
create_vpc_endpoint	Creates a VPC endpoint
create_vpc_endpoint_connection_notification	Creates a connection notification for a specified V
create_vpc_endpoint_service_configuration	Creates a VPC endpoint service to which service
create_vpc_peering_connection	Requests a VPC peering connection between two
create_vpn_connection	Creates a VPN connection between an existing v
create_vpn_connection_route	Creates a static route associated with a VPN con
create_vpn_gateway	Creates a virtual private gateway
delete_carrier_gateway	Deletes a carrier gateway
delete_client_vpn_endpoint	Deletes the specified Client VPN endpoint
delete_client_vpn_route	Deletes a route from a Client VPN endpoint
delete_coip_cidr	Deletes a range of customer-owned IP addresses
delete_coip_pool	Deletes a pool of customer-owned IP (CoIP) add
delete_customer_gateway	Deletes the specified customer gateway
delete_dhcp_options	Deletes the specified set of DHCP options
delete_egress_only_internet_gateway	Deletes an egress-only internet gateway
delete_fleets	Deletes the specified EC2 Fleets
delete_flow_logs	Deletes one or more flow logs
delete_fpga_image	Deletes the specified Amazon FPGA Image (AFI
delete_instance_connect_endpoint	Deletes the specified EC2 Instance Connect Endp
delete_instance_event_window	Deletes the specified event window

delete_internet_gateway	Deletes the specified internet gateway
delete_ipam	Delete an IPAM
delete_ipam_external_resource_verification_token	Delete a verification token
delete_ipam_pool	Delete an IPAM pool
delete_ipam_resource_discovery	Deletes an IPAM resource discovery
delete_ipam_scope	Delete the scope for an IPAM
delete_key_pair	Deletes the specified key pair, by removing the p
delete_launch_template	Deletes a launch template
delete_launch_template_versions	Deletes one or more versions of a launch templat
delete_local_gateway_route	Deletes the specified route from the specified loc
delete_local_gateway_route_table	Deletes a local gateway route table
delete_local_gateway_route_table_virtual_interface_group_association	Deletes a local gateway route table virtual interfa
delete_local_gateway_route_table_vpc_association	Deletes the specified association between a VPC
delete_managed_prefix_list	Deletes the specified managed prefix list
delete_nat_gateway	Deletes the specified NAT gateway
delete_network_acl	Deletes the specified network ACL
delete_network_acl_entry	Deletes the specified ingress or egress entry (rule
delete_network_insights_access_scope	Deletes the specified Network Access Scope
delete_network_insights_access_scope_analysis	Deletes the specified Network Access Scope ana
delete_network_insights_analysis	Deletes the specified network insights analysis
delete_network_insights_path	Deletes the specified path
delete_network_interface	Deletes the specified network interface
delete_network_interface_permission	Deletes a permission for a network interface
delete_placement_group	Deletes the specified placement group
delete_public_ipv_4_pool	Delete a public IPv4 pool
delete_queued_reserved_instances	Deletes the queued purchases for the specified R
delete_route	Deletes the specified route from the specified rou
delete_route_table	Deletes the specified route table
delete_security_group	Deletes a security group
delete_snapshot	Deletes the specified snapshot
delete_spot_datafeed_subscription	Deletes the data feed for Spot Instances
delete_subnet	Deletes the specified subnet
delete_subnet_cidr_reservation	Deletes a subnet CIDR reservation
delete_tags	Deletes the specified set of tags from the specifie
delete_traffic_mirror_filter	Deletes the specified Traffic Mirror filter
delete_traffic_mirror_filter_rule	Deletes the specified Traffic Mirror rule
delete_traffic_mirror_session	Deletes the specified Traffic Mirror session
delete_traffic_mirror_target	Deletes the specified Traffic Mirror target
delete_transit_gateway	Deletes the specified transit gateway
delete_transit_gateway_connect	Deletes the specified Connect attachment
delete_transit_gateway_connect_peer	Deletes the specified Connect peer
delete_transit_gateway_multicast_domain	Deletes the specified transit gateway multicast do
delete_transit_gateway_peering_attachment	Deletes a transit gateway peering attachment
delete_transit_gateway_policy_table	Deletes the specified transit gateway policy table
delete_transit_gateway_prefix_list_reference	Deletes a reference (route) to a prefix list in a spe
delete_transit_gateway_route	Deletes the specified route from the specified tra
delete_transit_gateway_route_table	Deletes the specified transit gateway route table
delete_transit_gateway_route_table_announcement	Advertises to the transit gateway that a transit ga

delete_transit_gateway_vpc_attachment	Deletes the specified VPC attachment
delete_verified_access_endpoint	Delete an Amazon Web Services Verified Access
delete_verified_access_group	Delete an Amazon Web Services Verified Access
delete_verified_access_instance	Delete an Amazon Web Services Verified Access
delete_verified_access_trust_provider	Delete an Amazon Web Services Verified Access
delete_volume	Deletes the specified EBS volume
delete_vpc	Deletes the specified VPC
delete_vpc_block_public_access_exclusion	Delete a VPC Block Public Access (BPA) exclus
delete_vpc_endpoint_connection_notifications	Deletes the specified VPC endpoint connection n
delete_vpc_endpoints	Deletes the specified VPC endpoints
delete_vpc_endpoint_service_configurations	Deletes the specified VPC endpoint service confi
delete_vpc_peering_connection	Deletes a VPC peering connection
delete_vpn_connection	Deletes the specified VPN connection
delete_vpn_connection_route	Deletes the specified static route associated with
delete_vpn_gateway	Deletes the specified virtual private gateway
deprovision_byoip_cidr	Releases the specified address range that you pro
deprovision_ipam_byoasn	Deprovisions your Autonomous System Number
deprovision_ipam_pool_cidr	Deprovision a CIDR provisioned from an IPAM
deprovision_public_ipv_4_pool_cidr	Deprovision a CIDR from a public IPv4 pool
deregister_image	Deregisters the specified AMI
deregister_instance_event_notification_attributes	Deregisters tag keys to prevent tags that have the
deregister_transit_gateway_multicast_group_members	Deregisters the specified members (network inter
deregister_transit_gateway_multicast_group_sources	Deregisters the specified sources (network interfa
describe_account_attributes	Describes attributes of your Amazon Web Servic
describe_addresses	Describes the specified Elastic IP addresses or al
describe_addresses_attribute	Describes the attributes of the specified Elastic I
describe_address_transfers	Describes an Elastic IP address transfer
describe_aggregate_id_format	Describes the longer ID format settings for all re
describe_availability_zones	Describes the Availability Zones, Local Zones, a
describe_aws_network_performance_metric_subscriptions	Describes the current Infrastructure Performance
describe_bundle_tasks	Describes the specified bundle tasks or all of you
describe_byoip_cidrs	Describes the IP address ranges that were specifi
describe_capacity_block_extension_history	Describes the events for the specified Capacity B
describe_capacity_block_extension_offerings	Describes Capacity Block extension offerings av
describe_capacity_block_offerings	Describes Capacity Block offerings available for
describe_capacity_reservation_billing_requests	Describes a request to assign the billing of the ur
describe_capacity_reservation_fleets	Describes one or more Capacity Reservation Flee
describe_capacity_reservations	Describes one or more of your Capacity Reserva
describe_carrier_gateways	Describes one or more of your carrier gateways
describe_classic_link_instances	This action is deprecated
describe_client_vpn_authorization_rules	Describes the authorization rules for a specified C
describe_client_vpn_connections	Describes active client connections and connecti
describe_client_vpn_endpoints	Describes one or more Client VPN endpoints in t
describe_client_vpn_routes	Describes the routes for the specified Client VPN
describe_client_vpn_target_networks	Describes the target networks associated with the
describe_coip_pools	Describes the specified customer-owned address
describe_conversion_tasks	Describes the specified conversion tasks or all yo
describe_customer_gateways	Describes one or more of your VPN customer ga

describe_declarative_policies_reports	Describes the metadata of an account status repo
describe_dhcp_options	Describes your DHCP option sets
describe_egress_only_internet_gateways	Describes your egress-only internet gateways
describe_elastic_gpus	Amazon Elastic Graphics reached end of life on
describe_export_image_tasks	Describes the specified export image tasks or all
describe_export_tasks	Describes the specified export instance tasks or a
describe_fast_launch_images	Describe details for Windows AMIs that are conf
describe_fast_snapshot_restores	Describes the state of fast snapshot restores for y
describe_fleet_history	Describes the events for the specified EC2 Fleet
describe_fleet_instances	Describes the running instances for the specified
describe_fleets	Describes the specified EC2 Fleet or all of your
describe_flow_logs	Describes one or more flow logs
describe_fpga_image_attribute	Describes the specified attribute of the specified
describe_fpga_images	Describes the Amazon FPGA Images (AFIs) ava
describe_host_reservation_offerings	Describes the Dedicated Host reservations that a
describe_host_reservations	Describes reservations that are associated with D
describe_hosts	Describes the specified Dedicated Hosts or all yo
describe_iam_instance_profile_associations	Describes your IAM instance profile associations
describe_identity_id_format	Describes the ID format settings for resources fo
describe_id_format	Describes the ID format settings for your resourc
describe_image_attribute	Describes the specified attribute of the specified
describe_images	Describes the specified images (AMIs, AKIs, and
describe_import_image_tasks	Displays details about an import virtual machine
describe_import_snapshot_tasks	Describes your import snapshot tasks
describe_instance_attribute	Describes the specified attribute of the specified
describe_instance_connect_endpoints	Describes the specified EC2 Instance Connect En
describe_instance_credit_specifications	Describes the credit option for CPU usage of the
describe_instance_event_notification_attributes	Describes the tag keys that are registered to appe
describe_instance_event_windows	Describes the specified event windows or all even
describe_instance_image_metadata	Describes the AMI that was used to launch an ins
describe_instances	Describes the specified instances or all instances
describe_instance_status	Describes the status of the specified instances or
describe_instance_topology	Describes a tree-based hierarchy that represents t
describe_instance_type_offerings	Lists the instance types that are offered for the sp
describe_instance_types	Describes the specified instance types
describe_internet_gateways	Describes your internet gateways
describe_ipam_byoasn	Describes your Autonomous System Numbers (A
describe_ipam_external_resource_verification_tokens	Describe verification tokens
describe_ipam_pools	Get information about your IPAM pools
describe_ipam_resource_discoveries	Describes IPAM resource discoveries
describe_ipam_resource_discovery_associations	Describes resource discovery association with an
describe_ipams	Get information about your IPAM pools
describe_ipam_scopes	Get information about your IPAM scopes
describe_ipv_6_pools	Describes your IPv6 address pools
describe_key_pairs	Describes the specified key pairs or all of your ke
describe_launch_templates	Describes one or more launch templates
describe_launch_template_versions	Describes one or more versions of a specified lau
describe_local_gateway_route_tables	Describes one or more local gateway route tables

describe_local_gateway_route_table_virtual_interface_group_associations	Describes the associations between virtual interfa
describe_local_gateway_route_table_vpc_associations	Describes the specified associations between VP
describe_local_gateways	Describes one or more local gateways
describe_local_gateway_virtual_interface_groups	Describes the specified local gateway virtual inte
describe_local_gateway_virtual_interfaces	Describes the specified local gateway virtual inte
describe_locked_snapshots	Describes the lock status for a snapshot
describe_mac_hosts	Describes the specified EC2 Mac Dedicated Hos
describe_managed_prefix_lists	Describes your managed prefix lists and any Am
describe_moving_addresses	This action is deprecated
describe_nat_gateways	Describes your NAT gateways
describe_network_acls	Describes your network ACLs
describe_network_insights_access_scope_analyses	Describes the specified Network Access Scope a
describe_network_insights_access_scopes	Describes the specified Network Access Scopes
describe_network_insights_analyses	Describes one or more of your network insights a
describe_network_insights_paths	Describes one or more of your paths
describe_network_interface_attribute	Describes a network interface attribute
describe_network_interface_permissions	Describes the permissions for your network inter
describe_network_interfaces	Describes the specified network interfaces or all
describe_placement_groups	Describes the specified placement groups or all c
describe_prefix_lists	Describes available Amazon Web Services servic
describe_principal_id_format	Describes the ID format settings for the root user
describe_public_ipv_4_pools	Describes the specified IPv4 address pools
describe_regions	Describes the Regions that are enabled for your a
describe_replace_root_volume_tasks	Describes a root volume replacement task
describe_reserved_instances	Describes one or more of the Reserved Instances
describe_reserved_instances_listings	Describes your account's Reserved Instance listi
describe_reserved_instances_modifications	Describes the modifications made to your Reserv
describe_reserved_instances_offerings	Describes Reserved Instance offerings that are av
describe_route_tables	Describes your route tables
describe_scheduled_instance_availability	Finds available schedules that meet the specified
describe_scheduled_instances	Describes the specified Scheduled Instances or al
describe_security_group_references	Describes the VPCs on the other side of a VPC p
describe_security_group_rules	Describes one or more of your security group rul
describe_security_groups	Describes the specified security groups or all of y
describe_security_group_vpc_associations	Describes security group VPC associations made
describe_snapshot_attribute	Describes the specified attribute of the specified s
describe_snapshots	Describes the specified EBS snapshots available
describe_snapshot_tier_status	Describes the storage tier status of one or more A
describe_spot_datafeed_subscription	Describes the data feed for Spot Instances
describe_spot_fleet_instances	Describes the running instances for the specified
describe_spot_fleet_request_history	Describes the events for the specified Spot Fleet
describe_spot_fleet_requests	Describes your Spot Fleet requests
describe_spot_instance_requests	Describes the specified Spot Instance requests
describe_spot_price_history	Describes the Spot price history
describe_stale_security_groups	Describes the stale security group rules for secur
describe_store_image_tasks	Describes the progress of the AMI store tasks
describe_subnets	Describes your subnets
describe_tags	Describes the specified tags for your EC2 resour

describe_traffic_mirror_filter_rules	Describe traffic mirror filters that determine the t
describe_traffic_mirror_filters	Describes one or more Traffic Mirror filters
describe_traffic_mirror_sessions	Describes one or more Traffic Mirror sessions
describe_traffic_mirror_targets	Information about one or more Traffic Mirror tar
describe_transit_gateway_attachments	Describes one or more attachments between reso
describe_transit_gateway_connect_peers	Describes one or more Connect peers
describe_transit_gateway_connects	Describes one or more Connect attachments
describe_transit_gateway_multicast_domains	Describes one or more transit gateway multicast
describe_transit_gateway_peering_attachments	Describes your transit gateway peering attachme
describe_transit_gateway_policy_tables	Describes one or more transit gateway route poli
describe_transit_gateway_route_table_announcements	Describes one or more transit gateway route tabl
describe_transit_gateway_route_tables	Describes one or more transit gateway route tabl
describe_transit_gateways	Describes one or more transit gateways
describe_transit_gateway_vpc_attachments	Describes one or more VPC attachments
describe_trunk_interface_associations	Describes one or more network interface trunk as
describe_verified_access_endpoints	Describes the specified Amazon Web Services V
describe_verified_access_groups	Describes the specified Verified Access groups
describe_verified_access_instance_logging_configurations	Describes the specified Amazon Web Services V
describe_verified_access_instances	Describes the specified Amazon Web Services V
describe_verified_access_trust_providers	Describes the specified Amazon Web Services V
describe_volume_attribute	Describes the specified attribute of the specified
describe_volumes	Describes the specified EBS volumes or all of yo
describe_volumes_modifications	Describes the most recent volume modification r
describe_volume_status	Describes the status of the specified volumes
describe_vpc_attribute	Describes the specified attribute of the specified
describe_vpc_block_public_access_exclusions	Describe VPC Block Public Access (BPA) exclu
describe_vpc_block_public_access_options	Describe VPC Block Public Access (BPA) optio
describe_vpc_classic_link	This action is deprecated
describe_vpc_classic_link_dns_support	This action is deprecated
describe_vpc_endpoint_associations	Describes the VPC resources, VPC endpoint serv
describe_vpc_endpoint_connection_notifications	Describes the connection notifications for VPC e
describe_vpc_endpoint_connections	Describes the VPC endpoint connections to your
describe_vpc_endpoints	Describes your VPC endpoints
describe_vpc_endpoint_service_configurations	Describes the VPC endpoint service configuratio
describe_vpc_endpoint_service_permissions	Describes the principals (service consumers) that
describe_vpc_endpoint_services	Describes available services to which you can cr
describe_vpc_peering_connections	Describes your VPC peering connections
describe_vpcs	Describes your VPCs
describe_vpn_connections	Describes one or more of your VPN connections
describe_vpn_gateways	Describes one or more of your virtual private gat
detach_classic_link_vpc	This action is deprecated
detach_internet_gateway	Detaches an internet gateway from a VPC, disabl
detach_network_interface	Detaches a network interface from an instance
detach_verified_access_trust_provider	Detaches the specified Amazon Web Services Ve
detach_volume	Detaches an EBS volume from an instance
detach_vpn_gateway	Detaches a virtual private gateway from a VPC
disable_address_transfer	Disables Elastic IP address transfer
disable_allowed_images_settings	Disables Allowed AMIs for your account in the s

disable_aws_network_performance_metric_subscription	Disables Infrastructure Performance metric subsc
disable_ebs_encryption_by_default	Disables EBS encryption by default for your acc
disable_fast_launch	Discontinue Windows fast launch for a Windows
disable_fast_snapshot_restores	Disables fast snapshot restores for the specified s
disable_image	Sets the AMI state to disabled and removes all la
disable_image_block_public_access	Disables block public access for AMIs at the acc
disable_image_deprecation	Cancels the deprecation of the specified AMI
disable_image_deregistration_protection	Disables deregistration protection for an AMI
disable_ipam_organization_admin_account	Disable the IPAM account
disable_serial_console_access	Disables access to the EC2 serial console of all i
disable_snapshot_block_public_access	Disables the block public access for snapshots se
disable_transit_gateway_route_table_propagation	Disables the specified resource attachment from
disable_vgw_route_propagation	Disables a virtual private gateway (VGW) from p
disable_vpc_classic_link	This action is deprecated
disable_vpc_classic_link_dns_support	This action is deprecated
disassociate_address	Disassociates an Elastic IP address from the insta
disassociate_capacity_reservation_billing_owner	Cancels a pending request to assign billing of the
disassociate_client_vpn_target_network	Disassociates a target network from the specified
disassociate_enclave_certificate_iam_role	Disassociates an IAM role from an Certificate M
disassociate_iam_instance_profile	Disassociates an IAM instance profile from a run
disassociate_instance_event_window	Disassociates one or more targets from an event
disassociate_ipam_byoasn	Remove the association between your Autonomo
disassociate_ipam_resource_discovery	Disassociates a resource discovery from an Amaz
disassociate_nat_gateway_address	Disassociates secondary Elastic IP addresses (EI
disassociate_route_table	Disassociates a subnet or gateway from a route ta
disassociate_security_group_vpc	Disassociates a security group from a VPC
disassociate_subnet_cidr_block	Disassociates a CIDR block from a subnet
disassociate_transit_gateway_multicast_domain	Disassociates the specified subnets from the trans
disassociate_transit_gateway_policy_table	Removes the association between an an attachme
disassociate_transit_gateway_route_table	Disassociates a resource attachment from a trans
disassociate_trunk_interface	Removes an association between a branch netwo
disassociate_vpc_cidr_block	Disassociates a CIDR block from a VPC
enable_address_transfer	Enables Elastic IP address transfer
enable_allowed_images_settings	Enables Allowed AMIs for your account in the s
enable_aws_network_performance_metric_subscription	Enables Infrastructure Performance subscriptions
enable_ebs_encryption_by_default	Enables EBS encryption by default for your acco
enable_fast_launch	When you enable Windows fast launch for a Win
enable_fast_snapshot_restores	Enables fast snapshot restores for the specified sn
enable_image	Re-enables a disabled AMI
enable_image_block_public_access	Enables block public access for AMIs at the acco
enable_image_deprecation	Enables deprecation of the specified AMI at the s
enable_image_deregistration_protection	Enables deregistration protection for an AMI
enable_ipam_organization_admin_account	Enable an Organizations member account as the
enable_reachability_analyzer_organization_sharing	Establishes a trust relationship between Reachab
enable_serial_console_access	Enables access to the EC2 serial console of all in
enable_snapshot_block_public_access	Enables or modifies the block public access for s
enable_transit_gateway_route_table_propagation	Enables the specified attachment to propagate ro
enable_vgw_route_propagation	Enables a virtual private gateway (VGW) to prop

enable_volume_io	Enables I/O operations for a volume that had I/O
enable_vpc_classic_link	This action is deprecated
enable_vpc_classic_link_dns_support	This action is deprecated
export_client_vpn_client_certificate_revocation_list	Downloads the client certificate revocation list fo
export_client_vpn_client_configuration	Downloads the contents of the Client VPN endpo
export_image	Exports an Amazon Machine Image (AMI) to a V
export_transit_gateway_routes	Exports routes from the specified transit gateway
export_verified_access_instance_client_configuration	Exports the client configuration for a Verified Ac
get_allowed_images_settings	Gets the current state of the Allowed AMIs settin
get_associated_enclave_certificate_iam_roles	Returns the IAM roles that are associated with th
get_associated_ipv_6_pool_cidrs	Gets information about the IPv6 CIDR block ass
get_aws_network_performance_data	Gets network performance data
get_capacity_reservation_usage	Gets usage information about a Capacity Reserva
get_coip_pool_usage	Describes the allocations from the specified custo
get_console_output	Gets the console output for the specified instance
get_console_screenshot	Retrieve a JPG-format screenshot of a running in
get_declarative_policies_report_summary	Retrieves a summary of the account status report
get_default_credit_specification	Describes the default credit option for CPU usag
get_ebs_default_kms_key_id	Describes the default KMS key for EBS encrypti
get_ebs_encryption_by_default	Describes whether EBS encryption by default is
get_flow_logs_integration_template	Generates a CloudFormation template that strean
get_groups_for_capacity_reservation	Lists the resource groups to which a Capacity Re
get_host_reservation_purchase_preview	Preview a reservation purchase with configuratio
get_image_block_public_access_state	Gets the current state of block public access for A
get_instance_metadata_defaults	Gets the default instance metadata service (IMDS
get_instance_tpm_ek_pub	Gets the public endorsement key associated with
get_instance_types_from_instance_requirements	Returns a list of instance types with the specified
get_instance_uefi_data	A binary representation of the UEFI variable stor
get_ipam_address_history	Retrieve historical information about a CIDR wit
get_ipam_discovered_accounts	Gets IPAM discovered accounts
get_ipam_discovered_public_addresses	Gets the public IP addresses that have been disco
get_ipam_discovered_resource_cidrs	Returns the resource CIDRs that are monitored a
get_ipam_pool_allocations	Get a list of all the CIDR allocations in an IPAM
get_ipam_pool_cidrs	Get the CIDRs provisioned to an IPAM pool
get_ipam_resource_cidrs	Returns resource CIDRs managed by IPAM in a
get_launch_template_data	Retrieves the configuration data of the specified i
get_managed_prefix_list_associations	Gets information about the resources that are ass
get_managed_prefix_list_entries	Gets information about the entries for a specified
get_network_insights_access_scope_analysis_findings	Gets the findings for the specified Network Acce
get_network_insights_access_scope_content	Gets the content for the specified Network Acces
get_password_data	Retrieves the encrypted administrator password f
get_reserved_instances_exchange_quote	Returns a quote and exchange information for ex
get_security_groups_for_vpc	Gets security groups that can be associated by th
get_serial_console_access_status	Retrieves the access status of your account to the
get_snapshot_block_public_access_state	Gets the current state of block public access for s
get_spot_placement_scores	Calculates the Spot placement score for a Region
get_subnet_cidr_reservations	Gets information about the subnet CIDR reservat
get_transit_gateway_attachment_propagations	Lists the route tables to which the specified resou

get_transit_gateway_multicast_domain_associations	Gets information about the associations for the tr
get_transit_gateway_policy_table_associations	Gets a list of the transit gateway policy table asso
get_transit_gateway_policy_table_entries	Returns a list of transit gateway policy table entri
get_transit_gateway_prefix_list_references	Gets information about the prefix list references
get_transit_gateway_route_table_associations	Gets information about the associations for the sp
get_transit_gateway_route_table_propagations	Gets information about the route table propagatio
get_verified_access_endpoint_policy	Get the Verified Access policy associated with th
get_verified_access_endpoint_targets	Gets the targets for the specified network CIDR e
get_verified_access_group_policy	Shows the contents of the Verified Access policy
get_vpn_connection_device_sample_configuration	Download an Amazon Web Services-provided sa
get_vpn_connection_device_types	Obtain a list of customer gateway devices for wh
get_vpn_tunnel_replacement_status	Get details of available tunnel endpoint maintena
import_client_vpn_client_certificate_revocation_list	Uploads a client certificate revocation list to the s
import_image	To import your virtual machines (VMs) with a co
import_instance	We recommend that you use the ImportImage AI
import_key_pair	Imports the public key from an RSA or ED25519
import_snapshot	Imports a disk into an EBS snapshot
import_volume	This API action supports only single-volume VM
list_images_in_recycle_bin	Lists one or more AMIs that are currently in the
list_snapshots_in_recycle_bin	Lists one or more snapshots that are currently in
lock_snapshot	Locks an Amazon EBS snapshot in either governa
modify_address_attribute	Modifies an attribute of the specified Elastic IP a
modify_availability_zone_group	Changes the opt-in status of the specified zone gr
modify_capacity_reservation	Modifies a Capacity Reservation's capacity, insta
modify_capacity_reservation_fleet	Modifies a Capacity Reservation Fleet
modify_client_vpn_endpoint	Modifies the specified Client VPN endpoint
modify_default_credit_specification	Modifies the default credit option for CPU usage
modify_ebs_default_kms_key_id	Changes the default KMS key for EBS encryptio
modify_fleet	Modifies the specified EC2 Fleet
modify_fpga_image_attribute	Modifies the specified attribute of the specified A
modify_hosts	Modify the auto-placement setting of a Dedicated
modify_identity_id_format	Modifies the ID format of a resource for a specifi
modify_id_format	Modifies the ID format for the specified resource
modify_image_attribute	Modifies the specified attribute of the specified A
modify_instance_attribute	Modifies the specified attribute of the specified in
modify_instance_capacity_reservation_attributes	Modifies the Capacity Reservation settings for a
modify_instance_cpu_options	By default, all vCPUs for the instance type are ac
modify_instance_credit_specification	Modifies the credit option for CPU usage on a ru
modify_instance_event_start_time	Modifies the start time for a scheduled Amazon I
modify_instance_event_window	Modifies the specified event window
modify_instance_maintenance_options	Modifies the recovery behavior of your instance t
modify_instance_metadata_defaults	Modifies the default instance metadata service (I
modify_instance_metadata_options	Modify the instance metadata parameters on a ru
modify_instance_network_performance_options	Change the configuration of the network perform
modify_instance_placement	Modifies the placement attributes for a specified
modify_ipam	Modify the configurations of an IPAM
modify_ipam_pool	Modify the configurations of an IPAM pool
modify_ipam_resource_cidr	Modify a resource CIDR

modify_ipam_resource_discovery	Modifies a resource discovery
modify_ipam_scope	Modify an IPAM scope
modify_launch_template	Modifies a launch template
modify_local_gateway_route	Modifies the specified local gateway route
modify_managed_prefix_list	Modifies the specified managed prefix list
modify_network_interface_attribute	Modifies the specified network interface attribute
modify_private_dns_name_options	Modifies the options for instance hostnames for t
modify_reserved_instances	Modifies the configuration of your Reserved Inst
modify_security_group_rules	Modifies the rules of a security group
modify_snapshot_attribute	Adds or removes permission settings for the spec
modify_snapshot_tier	Archives an Amazon EBS snapshot
modify_spot_fleet_request	Modifies the specified Spot Fleet request
modify_subnet_attribute	Modifies a subnet attribute
modify_traffic_mirror_filter_network_services	Allows or restricts mirroring network services
modify_traffic_mirror_filter_rule	Modifies the specified Traffic Mirror rule
modify_traffic_mirror_session	Modifies a Traffic Mirror session
modify_transit_gateway	Modifies the specified transit gateway
modify_transit_gateway_prefix_list_reference	Modifies a reference (route) to a prefix list in a sp
modify_transit_gateway_vpc_attachment	Modifies the specified VPC attachment
modify_verified_access_endpoint	Modifies the configuration of the specified Amaz
modify_verified_access_endpoint_policy	Modifies the specified Amazon Web Services Ve
modify_verified_access_group	Modifies the specified Amazon Web Services Ve
modify_verified_access_group_policy	Modifies the specified Amazon Web Services Ve
modify_verified_access_instance	Modifies the configuration of the specified Amaz
modify_verified_access_instance_logging_configuration	Modifies the logging configuration for the specifi
modify_verified_access_trust_provider	Modifies the configuration of the specified Amaz
modify_volume	You can modify several parameters of an existing
modify_volume_attribute	Modifies a volume attribute
modify_vpc_attribute	Modifies the specified attribute of the specified V
modify_vpc_block_public_access_exclusion	Modify VPC Block Public Access (BPA) exclusi
modify_vpc_block_public_access_options	Modify VPC Block Public Access (BPA) options
modify_vpc_endpoint	Modifies attributes of a specified VPC endpoint
modify_vpc_endpoint_connection_notification	Modifies a connection notification for VPC endp
modify_vpc_endpoint_service_configuration	Modifies the attributes of the specified VPC endp
modify_vpc_endpoint_service_payer_responsibility	Modifies the payer responsibility for your VPC e
modify_vpc_endpoint_service_permissions	Modifies the permissions for your VPC endpoint
modify_vpc_peering_connection_options	Modifies the VPC peering connection options on
modify_vpc_tenancy	Modifies the instance tenancy attribute of the spe
modify_vpn_connection	Modifies the customer gateway or the target gate
modify_vpn_connection_options	Modifies the connection options for your Site-to-
modify_vpn_tunnel_certificate	Modifies the VPN tunnel endpoint certificate
modify_vpn_tunnel_options	Modifies the options for a VPN tunnel in an Ama
monitor_instances	Enables detailed monitoring for a running instanc
move_address_to_vpc	This action is deprecated
move_byoip_cidr_to_ipam	Move a BYOIPv4 CIDR to IPAM from a public
move_capacity_reservation_instances	Move available capacity from a source Capacity
provision_byoip_cidr	Provisions an IPv4 or IPv6 address range for use
provision_ipam_byoasn	Provisions your Autonomous System Number (A

provision_ipam_pool_cidr	Provision a CIDR to an IPAM pool
provision_public_ipv_4_pool_cidr	Provision a CIDR to a public IPv4 pool
purchase_capacity_block	Purchase the Capacity Block for use with your ac
purchase_capacity_block_extension	Purchase the Capacity Block extension for use w
purchase_host_reservation	Purchase a reservation with configurations that m
purchase_reserved_instances_offering	Purchases a Reserved Instance for use with your
purchase_scheduled_instances	You can no longer purchase Scheduled Instances
reboot_instances	Requests a reboot of the specified instances
register_image	Registers an AMI
register_instance_event_notification_attributes	Registers a set of tag keys to include in schedule
register_transit_gateway_multicast_group_members	Registers members (network interfaces) with the
register_transit_gateway_multicast_group_sources	Registers sources (network interfaces) with the s
reject_capacity_reservation_billing_ownership	Rejects a request to assign billing of the availabl
reject_transit_gateway_multicast_domain_associations	Rejects a request to associate cross-account subn
reject_transit_gateway_peering_attachment	Rejects a transit gateway peering attachment requ
reject_transit_gateway_vpc_attachment	Rejects a request to attach a VPC to a transit gate
reject_vpc_endpoint_connections	Rejects VPC endpoint connection requests to you
reject_vpc_peering_connection	Rejects a VPC peering connection request
release_address	Releases the specified Elastic IP address
release_hosts	When you no longer want to use an On-Demand
release_ipam_pool_allocation	Release an allocation within an IPAM pool
replace_iam_instance_profile_association	Replaces an IAM instance profile for the specifie
replace_image_criteria_in_allowed_images_settings	Sets or replaces the criteria for Allowed AMIs
replace_network_acl_association	Changes which network ACL a subnet is associa
replace_network_acl_entry	Replaces an entry (rule) in a network ACL
replace_route	Replaces an existing route within a route table in
replace_route_table_association	Changes the route table associated with a given s
replace_transit_gateway_route	Replaces the specified route in the specified trans
replace_vpn_tunnel	Trigger replacement of specified VPN tunnel
report_instance_status	Submits feedback about the status of an instance
request_spot_fleet	Creates a Spot Fleet request
request_spot_instances	Creates a Spot Instance request
reset_address_attribute	Resets the attribute of the specified IP address
reset_ebs_default_kms_key_id	Resets the default KMS key for EBS encryption
reset_fpga_image_attribute	Resets the specified attribute of the specified Am
reset_image_attribute	Resets an attribute of an AMI to its default value
reset_instance_attribute	Resets an attribute of an instance to its default va
reset_network_interface_attribute	Resets a network interface attribute
reset_snapshot_attribute	Resets permission settings for the specified snaps
restore_address_to_classic	This action is deprecated
restore_image_from_recycle_bin	Restores an AMI from the Recycle Bin
restore_managed_prefix_list_version	Restores the entries from a previous version of a
restore_snapshot_from_recycle_bin	Restores a snapshot from the Recycle Bin
restore_snapshot_tier	Restores an archived Amazon EBS snapshot for
revoke_client_vpn_ingress	Removes an ingress authorization rule from a Cl
revoke_security_group_egress	Removes the specified outbound (egress) rules fr
revoke_security_group_ingress	Removes the specified inbound (ingress) rules fr
run_instances	Launches the specified number of instances using

run_scheduled_instances	Launches the specified Scheduled Instances
search_local_gateway_routes	Searches for routes in the specified local gateway
search_transit_gateway_multicast_groups	Searches one or more transit gateway multicast g
search_transit_gateway_routes	Searches for routes in the specified transit gatewa
send_diagnostic_interrupt	Sends a diagnostic interrupt to the specified Ama
start_declarative_policies_report	Generates an account status report
start_instances	Starts an Amazon EBS-backed instance that you
start_network_insights_access_scope_analysis	Starts analyzing the specified Network Access Sc
start_network_insights_analysis	Starts analyzing the specified path
start_vpc_endpoint_service_private_dns_verification	Initiates the verification process to prove that the
stop_instances	Stops an Amazon EBS-backed instance
terminate_client_vpn_connections	Terminates active Client VPN endpoint connecti
terminate_instances	Shuts down the specified instances
unassign_ipv_6_addresses	Unassigns the specified IPv6 addresses or Prefix
unassign_private_ip_addresses	Unassigns the specified secondary private IP add
unassign_private_nat_gateway_address	Unassigns secondary private IPv4 addresses fron
unlock_snapshot	Unlocks a snapshot that is locked in governance
unmonitor_instances	Disables detailed monitoring for a running instan
update_security_group_rule_descriptions_egress	Updates the description of an egress (outbound) s
update_security_group_rule_descriptions_ingress	Updates the description of an ingress (inbound) s
withdraw_byoip_cidr	Stops advertising an address range that is provisi

Examples

```
## Not run:
svc <- ec2()
# This example allocates an Elastic IP address to use with an instance in
# a VPC.
svc$allocate_address(
  Domain = "vpc"
)

## End(Not run)
```

`ec2instanceconnect` *AWS EC2 Instance Connect*

Description

This is the *Amazon EC2 Instance Connect API Reference*. It provides descriptions, syntax, and usage examples for each of the actions for Amazon EC2 Instance Connect. Amazon EC2 Instance Connect enables system administrators to publish one-time use SSH public keys to EC2, providing users a simple and secure way to connect to their instances.

To view the Amazon EC2 Instance Connect content in the *Amazon EC2 User Guide*, see Connect to your Linux instance using EC2 Instance Connect.

For Amazon EC2 APIs, see the Amazon EC2 API Reference.

Usage

```
ec2instanceconnect(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

```
svc <- ec2instanceconnect(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

send_serial_console_ssh_public_key	Pushes an SSH public key to the specified EC2 instance
send_ssh_public_key	Pushes an SSH public key to the specified EC2 instance for use by the specified user

Examples

```
## Not run:
svc <- ec2instanceconnect()
# The following example pushes a sample SSH public key to the EC2 instance
# i-abcd1234 in AZ us-west-2b for use by the instance OS user ec2-user.
svc$send_ssh_public_key(
  AvailabilityZone = "us-west-2a",
  InstanceId = "i-abcd1234",
```

```
  InstanceOSUser = "ec2-user",
  SSHPublicKey = "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC3FlHqj2eqCdrGHuA6d..."
)

## End(Not run)
```

---

ecr *Amazon Elastic Container Registry*

Description

Amazon Elastic Container Registry (Amazon ECR) is a managed container image registry service. Customers can use the familiar Docker CLI, or their preferred client, to push, pull, and manage images. Amazon ECR provides a secure, scalable, and reliable registry for your Docker or Open Container Initiative (OCI) images. Amazon ECR supports private repositories with resource-based permissions using IAM so that specific users or Amazon EC2 instances can access repositories and images.

Amazon ECR has service endpoints in each supported Region. For more information, see Amazon ECR endpoints in the *Amazon Web Services General Reference*.

Usage

```
ecr(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
    region = "string"
  )
```

Operations

batch_check_layer_availability	Checks the availability of one or more image layers in a repository
batch_delete_image	Deletes a list of specified images within a repository
batch_get_image	Gets detailed information for an image
batch_get_repository_scanning_configuration	Gets the scanning configuration for one or more repositories
complete_layer_upload	Informs Amazon ECR that the image layer upload has completed for a specifi
create_pull_through_cache_rule	Creates a pull through cache rule
create_repository	Creates a repository
create_repository_creation_template	Creates a repository creation template
delete_lifecycle_policy	Deletes the lifecycle policy associated with the specified repository
delete_pull_through_cache_rule	Deletes a pull through cache rule
delete_registry_policy	Deletes the registry permissions policy
delete_repository	Deletes a repository
delete_repository_creation_template	Deletes a repository creation template
delete_repository_policy	Deletes the repository policy associated with the specified repository
describe_image_replication_status	Returns the replication status for a specified image
describe_images	Returns metadata about the images in a repository
describe_image_scan_findings	Returns the scan findings for the specified image
describe_pull_through_cache_rules	Returns the pull through cache rules for a registry
describe_registry	Describes the settings for a registry
describe_repositories	Describes image repositories in a registry
describe_repository_creation_templates	Returns details about the repository creation templates in a registry
get_account_setting	Retrieves the account setting value for the specified setting name
get_authorization_token	Retrieves an authorization token
get_download_url_for_layer	Retrieves the pre-signed Amazon S3 download URL corresponding to an ima
get_lifecycle_policy	Retrieves the lifecycle policy for the specified repository
get_lifecycle_policy_preview	Retrieves the results of the lifecycle policy preview request for the specified r
get_registry_policy	Retrieves the permissions policy for a registry
get_registry_scanning_configuration	Retrieves the scanning configuration for a registry
get_repository_policy	Retrieves the repository policy for the specified repository
initiate_layer_upload	Notifies Amazon ECR that you intend to upload an image layer
list_images	Lists all the image IDs for the specified repository
list_tags_for_resource	List the tags for an Amazon ECR resource
put_account_setting	Allows you to change the basic scan type version or registry policy scope
put_image	Creates or updates the image manifest and tags associated with an image
put_image_scanning_configuration	The PutImageScanningConfiguration API is being deprecated, in favor of spe
put_image_tag_mutability	Updates the image tag mutability settings for the specified repository
put_lifecycle_policy	Creates or updates the lifecycle policy for the specified repository
put_registry_policy	Creates or updates the permissions policy for your registry
put_registry_scanning_configuration	Creates or updates the scanning configuration for your private registry
put_replication_configuration	Creates or updates the replication configuration for a registry
set_repository_policy	Applies a repository policy to the specified repository to control access permi
start_image_scan	Starts a basic image vulnerability scan
start_lifecycle_policy_preview	Starts a preview of a lifecycle policy for the specified repository

tag_resource	Adds specified tags to a resource with the specified ARN
untag_resource	Deletes specified tags from a resource
update_pull_through_cache_rule	Updates an existing pull through cache rule
update_repository_creation_template	Updates an existing repository creation template
upload_layer_part	Uploads an image layer part to Amazon ECR
validate_pull_through_cache_rule	Validates an existing pull through cache rule for an upstream registry that req

Examples

```
## Not run:
svc <- ecr()
# This example deletes images with the tags precise and trusty in a
# repository called ubuntu in the default registry for an account.
svc$batch_delete_image(
  imageIds = list(
    list(
      imageTag = "precise"
    )
  ),
  repositoryName = "ubuntu"
)

## End(Not run)
```

`ecrpublic` *Amazon Elastic Container Registry Public*

Description

Amazon Elastic Container Registry Public (Amazon ECR Public) is a managed container image registry service. Amazon ECR provides both public and private registries to host your container images. You can use the Docker CLI or your preferred client to push, pull, and manage images. Amazon ECR provides a secure, scalable, and reliable registry for your Docker or Open Container Initiative (OCI) images. Amazon ECR supports public repositories with this API. For information about the Amazon ECR API for private repositories, see Amazon Elastic Container Registry API Reference.

Usage

```
ecrpublic(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)
```

Operations

batch_check_layer_availability	Checks the availability of one or more image layers that are within a repository in a public re
batch_delete_image	Deletes a list of specified images that are within a repository in a public registry
complete_layer_upload	Informs Amazon ECR that the image layer upload is complete for a specified public registry,
create_repository	Creates a repository in a public registry
delete_repository	Deletes a repository in a public registry
delete_repository_policy	Deletes the repository policy that's associated with the specified repository
describe_images	Returns metadata that's related to the images in a repository in a public registry
describe_image_tags	Returns the image tag details for a repository in a public registry
describe_registries	Returns details for a public registry
describe_repositories	Describes repositories that are in a public registry
get_authorization_token	Retrieves an authorization token
get_registry_catalog_data	Retrieves catalog metadata for a public registry
get_repository_catalog_data	Retrieve catalog metadata for a repository in a public registry
get_repository_policy	Retrieves the repository policy for the specified repository
initiate_layer_upload	Notifies Amazon ECR that you intend to upload an image layer
list_tags_for_resource	List the tags for an Amazon ECR Public resource
put_image	Creates or updates the image manifest and tags that are associated with an image
put_registry_catalog_data	Create or update the catalog data for a public registry
put_repository_catalog_data	Creates or updates the catalog data for a repository in a public registry
set_repository_policy	Applies a repository policy to the specified public repository to control access permissions

tag_resource	Associates the specified tags to a resource with the specified resourceArn
untag_resource	Deletes specified tags from a resource
upload_layer_part	Uploads an image layer part to Amazon ECR

Examples

```
## Not run:
svc <- ecrpublic()
svc$batch_check_layer_availability(
  Foo = 123
)

## End(Not run)
```

`ecs` *Amazon EC2 Container Service*

Description

Amazon Elastic Container Service

Amazon Elastic Container Service (Amazon ECS) is a highly scalable, fast, container management service. It makes it easy to run, stop, and manage Docker containers. You can host your cluster on a serverless infrastructure that's managed by Amazon ECS by launching your services or tasks on Fargate. For more control, you can host your tasks on a cluster of Amazon Elastic Compute Cloud (Amazon EC2) or External (on-premises) instances that you manage.

Amazon ECS makes it easy to launch and stop container-based applications with simple API calls. This makes it easy to get the state of your cluster from a centralized service, and gives you access to many familiar Amazon EC2 features.

You can use Amazon ECS to schedule the placement of containers across your cluster based on your resource needs, isolation policies, and availability requirements. With Amazon ECS, you don't need to operate your own cluster management and configuration management systems. You also don't need to worry about scaling your management infrastructure.

Usage

```
ecs(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

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

- **credentials**:
  - **creds**:
    - **access_key_id**: AWS access key ID

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

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
      region = "string",
      close_connection = "logical",
      timeout = "numeric",
      s3_force_path_style = "logical",
      sts_regional_endpoint = "string"
    ),
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
  )
```

Operations

create_capacity_provider	Creates a new capacity provider
create_cluster	Creates a new Amazon ECS cluster
create_service	Runs and maintains your desired number of tasks from a specified task definition
create_task_set	Create a task set in the specified cluster and service
delete_account_setting	Disables an account setting for a specified user, role, or the root user for an account
delete_attributes	Deletes one or more custom attributes from an Amazon ECS resource
delete_capacity_provider	Deletes the specified capacity provider
delete_cluster	Deletes the specified cluster
delete_service	Deletes a specified service within a cluster
delete_task_definitions	Deletes one or more task definitions
delete_task_set	Deletes a specified task set within a service
deregister_container_instance	Deregisters an Amazon ECS container instance from the specified cluster
deregister_task_definition	Deregisters the specified task definition by family and revision
describe_capacity_providers	Describes one or more of your capacity providers
describe_clusters	Describes one or more of your clusters
describe_container_instances	Describes one or more container instances
describe_service_deployments	Describes one or more of your service deployments
describe_service_revisions	Describes one or more service revisions
describe_services	Describes the specified services running in your cluster
describe_task_definition	Describes a task definition
describe_tasks	Describes a specified task or tasks
describe_task_sets	Describes the task sets in the specified cluster and service
discover_poll_endpoint	This action is only used by the Amazon ECS agent, and it is not intended for use outside
execute_command	Runs a command remotely on a container within a task
get_task_protection	Retrieves the protection status of tasks in an Amazon ECS service
list_account_settings	Lists the account settings for a specified principal
list_attributes	Lists the attributes for Amazon ECS resources within a specified target type and cluster

list_clusters	Returns a list of existing clusters
list_container_instances	Returns a list of container instances in a specified cluster
list_service_deployments	This operation lists all the service deployments that meet the specified filter criteria
list_services	Returns a list of services
list_services_by_namespace	This operation lists all of the services that are associated with a Cloud Map namespace
list_tags_for_resource	List the tags for an Amazon ECS resource
list_task_definition_families	Returns a list of task definition families that are registered to your account
list_task_definitions	Returns a list of task definitions that are registered to your account
list_tasks	Returns a list of tasks
put_account_setting	Modifies an account setting
put_account_setting_default	Modifies an account setting for all users on an account for whom no individual account se
put_attributes	Create or update an attribute on an Amazon ECS resource
put_cluster_capacity_providers	Modifies the available capacity providers and the default capacity provider strategy for a
register_container_instance	This action is only used by the Amazon ECS agent, and it is not intended for use outside
register_task_definition	Registers a new task definition from the supplied family and containerDefinitions
run_task	Starts a new task using the specified task definition
start_task	Starts a new task from the specified task definition on the specified container instance or i
stop_task	Stops a running task
submit_attachment_state_changes	This action is only used by the Amazon ECS agent, and it is not intended for use outside
submit_container_state_change	This action is only used by the Amazon ECS agent, and it is not intended for use outside
submit_task_state_change	This action is only used by the Amazon ECS agent, and it is not intended for use outside
tag_resource	Associates the specified tags to a resource with the specified resourceArn
untag_resource	Deletes specified tags from a resource
update_capacity_provider	Modifies the parameters for a capacity provider
update_cluster	Updates the cluster
update_cluster_settings	Modifies the settings to use for a cluster
update_container_agent	Updates the Amazon ECS container agent on a specified container instance
update_container_instances_state	Modifies the status of an Amazon ECS container instance
update_service	Modifies the parameters of a service
update_service_primary_task_set	Modifies which task set in a service is the primary task set
update_task_protection	Updates the protection status of a task
update_task_set	Modifies a task set

Examples

```
## Not run:
svc <- ecs()
# This example creates a cluster in your default region.
svc$create_cluster(
  clusterName = "my_cluster"
)

## End(Not run)
```

eks *Amazon Elastic Kubernetes Service*

Description

Amazon Elastic Kubernetes Service (Amazon EKS) is a managed service that makes it easy for you to run Kubernetes on Amazon Web Services without needing to setup or maintain your own Kubernetes control plane. Kubernetes is an open-source system for automating the deployment, scaling, and management of containerized applications.

Amazon EKS runs up-to-date versions of the open-source Kubernetes software, so you can use all the existing plugins and tooling from the Kubernetes community. Applications running on Amazon EKS are fully compatible with applications running on any standard Kubernetes environment, whether running in on-premises data centers or public clouds. This means that you can easily migrate any standard Kubernetes application to Amazon EKS without any code modification required.

Usage

```
eks(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

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

- **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

Operations

associate_access_policy	Associates an access policy and its scope to an access entry
associate_encryption_config	Associates an encryption configuration to an existing cluster
associate_identity_provider_config	Associates an identity provider configuration to a cluster
create_access_entry	Creates an access entry
create_addon	Creates an Amazon EKS add-on
create_cluster	Creates an Amazon EKS control plane
create_eks_anywhere_subscription	Creates an EKS Anywhere subscription
create_fargate_profile	Creates an Fargate profile for your Amazon EKS cluster
create_nodegroup	Creates a managed node group for an Amazon EKS cluster
create_pod_identity_association	Creates an EKS Pod Identity association between a service account in an Amazon EK
delete_access_entry	Deletes an access entry
delete_addon	Deletes an Amazon EKS add-on
delete_cluster	Deletes an Amazon EKS cluster control plane
delete_eks_anywhere_subscription	Deletes an expired or inactive subscription
delete_fargate_profile	Deletes an Fargate profile
delete_nodegroup	Deletes a managed node group
delete_pod_identity_association	Deletes a EKS Pod Identity association
deregister_cluster	Deregisters a connected cluster to remove it from the Amazon EKS control plane
describe_access_entry	Describes an access entry
describe_addon	Describes an Amazon EKS add-on
describe_addon_configuration	Returns configuration options
describe_addon_versions	Describes the versions for an add-on
describe_cluster	Describes an Amazon EKS cluster
describe_cluster_versions	Lists available Kubernetes versions for Amazon EKS clusters
describe_eks_anywhere_subscription	Returns descriptive information about a subscription
describe_fargate_profile	Describes an Fargate profile
describe_identity_provider_config	Describes an identity provider configuration
describe_insight	Returns details about an insight that you specify using its ID
describe_nodegroup	Describes a managed node group
describe_pod_identity_association	Returns descriptive information about an EKS Pod Identity association
describe_update	Describes an update to an Amazon EKS resource
disassociate_access_policy	Disassociates an access policy from an access entry
disassociate_identity_provider_config	Disassociates an identity provider configuration from a cluster
list_access_entries	Lists the access entries for your cluster
list_access_policies	Lists the available access policies
list_addons	Lists the installed add-ons
list_associated_access_policies	Lists the access policies associated with an access entry
list_clusters	Lists the Amazon EKS clusters in your Amazon Web Services account in the specifie
list_eks_anywhere_subscriptions	Displays the full description of the subscription
list_fargate_profiles	Lists the Fargate profiles associated with the specified cluster in your Amazon Web S
list_identity_provider_configs	Lists the identity provider configurations for your cluster
list_insights	Returns a list of all insights checked for against the specified cluster
list_nodegroups	Lists the managed node groups associated with the specified cluster in your Amazon
list_pod_identity_associations	List the EKS Pod Identity associations in a cluster
list_tags_for_resource	List the tags for an Amazon EKS resource
list_updates	Lists the updates associated with an Amazon EKS resource in your Amazon Web Ser
register_cluster	Connects a Kubernetes cluster to the Amazon EKS control plane
tag_resource	Associates the specified tags to an Amazon EKS resource with the specified resource

untag_resource	Deletes specified tags from an Amazon EKS resource
update_access_entry	Updates an access entry
update_addon	Updates an Amazon EKS add-on
update_cluster_config	Updates an Amazon EKS cluster configuration
update_cluster_version	Updates an Amazon EKS cluster to the specified Kubernetes version
update_eks_anywhere_subscription	Update an EKS Anywhere Subscription
update_nodegroup_config	Updates an Amazon EKS managed node group configuration
update_nodegroup_version	Updates the Kubernetes version or AMI version of an Amazon EKS managed node g
update_pod_identity_association	Updates a EKS Pod Identity association

Examples

```
## Not run:
svc <- eks()
# The following example creates an Amazon EKS cluster called prod.
svc$create_cluster(
  version = "1.10",
  name = "prod",
  clientRequestToken = "1d2129a1-3d38-460a-9756-e5b91fddb951",
  resourcesVpcConfig = list(
    securityGroupIds = list(
      "sg-6979fe18"
    ),
    subnetIds = list(
      "subnet-6782e71e",
      "subnet-e7e761ac"
    )
  ),
  roleArn = "arn:aws:iam::012345678910:role/eks-service-role-AWSServiceRole..."
)

## End(Not run)
```

`elasticbeanstalk` *AWS Elastic Beanstalk*

Description

AWS Elastic Beanstalk makes it easy for you to create, deploy, and manage scalable, fault-tolerant applications running on the Amazon Web Services cloud.

For more information about this product, go to the AWS Elastic Beanstalk details page. The location of the latest AWS Elastic Beanstalk WSDL is `https://elasticbeanstalk.s3.amazonaws.com/doc/2010-12-01/AWSElasticBeanstalk.wsdl`. To install the Software Development Kits (SDKs), Integrated Development Environment (IDE) Toolkits, and command line tools that enable you to access the API, go to Tools for Amazon Web Services.

Endpoints

For a list of region-specific endpoints that AWS Elastic Beanstalk supports, go to Regions and Endpoints in the *Amazon Web Services Glossary*.

Usage

```
elasticbeanstalk(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

Operations

abort_environment_update	Cancels in-progress environment configuration update or application versio
apply_environment_managed_action	Applies a scheduled managed action immediately
associate_environment_operations_role	Add or change the operations role used by an environment
check_dns_availability	Checks if the specified CNAME is available
compose_environments	Create or update a group of environments that each run a separate compone
create_application	Creates an application that has one configuration template named default ar
create_application_version	Creates an application version for the specified application
create_configuration_template	Creates an AWS Elastic Beanstalk configuration template, associated with a

create_environment	Launches an AWS Elastic Beanstalk environment for the specified applicati
create_platform_version	Create a new version of your custom platform
create_storage_location	Creates a bucket in Amazon S3 to store application versions, logs, and othe
delete_application	Deletes the specified application along with all associated versions and con
delete_application_version	Deletes the specified version from the specified application
delete_configuration_template	Deletes the specified configuration template
delete_environment_configuration	Deletes the draft configuration associated with the running environment
delete_platform_version	Deletes the specified version of a custom platform
describe_account_attributes	Returns attributes related to AWS Elastic Beanstalk that are associated with
describe_applications	Returns the descriptions of existing applications
describe_application_versions	Retrieve a list of application versions
describe_configuration_options	Describes the configuration options that are used in a particular configuratio
describe_configuration_settings	Returns a description of the settings for the specified configuration set, that
describe_environment_health	Returns information about the overall health of the specified environment
describe_environment_managed_action_history	Lists an environment's completed and failed managed actions
describe_environment_managed_actions	Lists an environment's upcoming and in-progress managed actions
describe_environment_resources	Returns AWS resources for this environment
describe_environments	Returns descriptions for existing environments
describe_events	Returns list of event descriptions matching criteria up to the last 6 weeks
describe_instances_health	Retrieves detailed information about the health of instances in your AWS E
describe_platform_version	Describes a platform version
disassociate_environment_operations_role	Disassociate the operations role from an environment
list_available_solution_stacks	Returns a list of the available solution stack names, with the public version
list_platform_branches	Lists the platform branches available for your account in an AWS Region
list_platform_versions	Lists the platform versions available for your account in an AWS Region
list_tags_for_resource	Return the tags applied to an AWS Elastic Beanstalk resource
rebuild_environment	Deletes and recreates all of the AWS resources (for example: the Auto Scal
request_environment_info	Initiates a request to compile the specified type of information of the deploy
restart_app_server	Causes the environment to restart the application container server running c
retrieve_environment_info	Retrieves the compiled information from a RequestEnvironmentInfo reques
swap_environment_cnam_es	Swaps the CNAMEs of two environments
terminate_environment	Terminates the specified environment
update_application	Updates the specified application to have the specified properties
update_application_resource_lifecycle	Modifies lifecycle settings for an application
update_application_version	Updates the specified application version to have the specified properties
update_configuration_template	Updates the specified configuration template to have the specified propertie
update_environment	Updates the environment description, deploys a new application version, up
update_tags_for_resource	Update the list of tags applied to an AWS Elastic Beanstalk resource
validate_configuration_settings	Takes a set of configuration settings and either a configuration template or e

Examples

```
## Not run:
svc <- elasticbeanstalk()
# The following code aborts a running application version deployment for
# an environment named my-env:
svc$abort_environment_update(
```

```
  EnvironmentName = "my-env"
)

## End(Not run)
```

`emrcontainers` *Amazon EMR Containers*

Description

Amazon EMR on EKS provides a deployment option for Amazon EMR that allows you to run open-source big data frameworks on Amazon Elastic Kubernetes Service (Amazon EKS). With this deployment option, you can focus on running analytics workloads while Amazon EMR on EKS builds, configures, and manages containers for open-source applications. For more information about Amazon EMR on EKS concepts and tasks, see What is Amazon EMR on EKS.

Amazon EMR containers is the API name for Amazon EMR on EKS. The `emr-containers` prefix is used in the following scenarios:

- It is the prefix in the CLI commands for Amazon EMR on EKS. For example, `aws emr-containers start-job-run`.
- It is the prefix before IAM policy actions for Amazon EMR on EKS. For example, `"Action": [ "emr-containers:Sta` For more information, see Policy actions for Amazon EMR on EKS.
- It is the prefix used in Amazon EMR on EKS service endpoints. For example, `emr-containers.us-east-2.amazonaws` For more information, see Amazon EMR on EKSService Endpoints.

Usage

```
emrcontainers(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

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

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

- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-e html

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

cancel_job_run	Cancels a job run
create_job_template	Creates a job template
create_managed_endpoint	Creates a managed endpoint
create_security_configuration	Creates a security configuration
create_virtual_cluster	Creates a virtual cluster
delete_job_template	Deletes a job template
delete_managed_endpoint	Deletes a managed endpoint
delete_virtual_cluster	Deletes a virtual cluster
describe_job_run	Displays detailed information about a job run
describe_job_template	Displays detailed information about a specified job template
describe_managed_endpoint	Displays detailed information about a managed endpoint
describe_security_configuration	Displays detailed information about a specified security configuration
describe_virtual_cluster	Displays detailed information about a specified virtual cluster
get_managed_endpoint_session_credentials	Generate a session token to connect to a managed endpoint
list_job_runs	Lists job runs based on a set of parameters
list_job_templates	Lists job templates based on a set of parameters
list_managed_endpoints	Lists managed endpoints based on a set of parameters
list_security_configurations	Lists security configurations based on a set of parameters
list_tags_for_resource	Lists the tags assigned to the resources
list_virtual_clusters	Lists information about the specified virtual cluster
start_job_run	Starts a job run
tag_resource	Assigns tags to resources
untag_resource	Removes tags from resources

Examples

```
## Not run:
svc <- emrcontainers()
svc$cancel_job_run(
  Foo = 123
)
```

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

`emrserverless` *EMR Serverless*

Description

Amazon EMR Serverless is a new deployment option for Amazon EMR. Amazon EMR Serverless provides a serverless runtime environment that simplifies running analytics applications using the latest open source frameworks such as Apache Spark and Apache Hive. With Amazon EMR Serverless, you don't have to configure, optimize, secure, or operate clusters to run applications with these frameworks.

The API reference to Amazon EMR Serverless is `emr-serverless`. The `emr-serverless` prefix is used in the following scenarios:

- It is the prefix in the CLI commands for Amazon EMR Serverless. For example, `aws emr-serverless start-job-run`
- It is the prefix before IAM policy actions for Amazon EMR Serverless. For example, `"Action": ["emr-serverless:S`. For more information, see Policy actions for Amazon EMR Serverless.
- It is the prefix used in Amazon EMR Serverless service endpoints. For example, `emr-serverless.us-east-2.amazona`

Usage

```
emrserverless(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

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

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

- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-e html

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

cancel_job_run	Cancels a job run
create_application	Creates an application
delete_application	Deletes an application
get_application	Displays detailed information about a specified application
get_dashboard_for_job_run	Creates and returns a URL that you can use to access the application UIs for a job run
get_job_run	Displays detailed information about a job run
list_applications	Lists applications based on a set of parameters
list_job_run_attempts	Lists all attempt of a job run
list_job_runs	Lists job runs based on a set of parameters
list_tags_for_resource	Lists the tags assigned to the resources
start_application	Starts a specified application and initializes initial capacity if configured
start_job_run	Starts a job run
stop_application	Stops a specified application and releases initial capacity if configured
tag_resource	Assigns tags to resources
untag_resource	Removes tags from resources
update_application	Updates a specified application

Examples

```
## Not run:
svc <- emrserverless()
svc$cancel_job_run(
  Foo = 123
)

## End(Not run)
```

---

imagebuilder *EC2 Image Builder*

Description

EC2 Image Builder is a fully managed Amazon Web Services service that makes it easier to automate the creation, management, and deployment of customized, secure, and up-to-date "golden" server images that are pre-installed and pre-configured with software and settings to meet specific IT standards.

Usage

```
imagebuilder(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

Operations

cancel_image_creation	CancelImageCreation cancels the creation of Image
cancel_lifecycle_execution	Cancel a specific image lifecycle policy runtime instance
create_component	Creates a new component that can be used to build, validate, test, and assess your ima
create_container_recipe	Creates a new container recipe
create_distribution_configuration	Creates a new distribution configuration
create_image	Creates a new image
create_image_pipeline	Creates a new image pipeline
create_image_recipe	Creates a new image recipe

create_infrastructure_configuration	Creates a new infrastructure configuration
create_lifecycle_policy	Create a lifecycle policy resource
create_workflow	Create a new workflow or a new version of an existing workflow
delete_component	Deletes a component build version
delete_container_recipe	Deletes a container recipe
delete_distribution_configuration	Deletes a distribution configuration
delete_image	Deletes an Image Builder image resource
delete_image_pipeline	Deletes an image pipeline
delete_image_recipe	Deletes an image recipe
delete_infrastructure_configuration	Deletes an infrastructure configuration
delete_lifecycle_policy	Delete the specified lifecycle policy resource
delete_workflow	Deletes a specific workflow resource
get_component	Gets a component object
get_component_policy	Gets a component policy
get_container_recipe	Retrieves a container recipe
get_container_recipe_policy	Retrieves the policy for a container recipe
get_distribution_configuration	Gets a distribution configuration
get_image	Gets an image
get_image_pipeline	Gets an image pipeline
get_image_policy	Gets an image policy
get_image_recipe	Gets an image recipe
get_image_recipe_policy	Gets an image recipe policy
get_infrastructure_configuration	Gets an infrastructure configuration
get_lifecycle_execution	Get the runtime information that was logged for a specific runtime instance of the life
get_lifecycle_policy	Get details for the specified image lifecycle policy
get_marketplace_resource	Verify the subscription and perform resource dependency checks on the requested Ar
get_workflow	Get a workflow resource object
get_workflow_execution	Get the runtime information that was logged for a specific runtime instance of the wo
get_workflow_step_execution	Get the runtime information that was logged for a specific runtime instance of the wo
import_component	Imports a component and transforms its data into a component document
import_disk_image	Import a Windows operating system image from a verified Microsoft ISO disk file
import_vm_image	When you export your virtual machine (VM) from its virtualization environment, tha
list_component_build_versions	Returns the list of component build versions for the specified component version Am
list_components	Returns the list of components that can be filtered by name, or by using the listed filt
list_container_recipes	Returns a list of container recipes
list_distribution_configurations	Returns a list of distribution configurations
list_image_build_versions	Returns a list of image build versions
list_image_packages	List the Packages that are associated with an Image Build Version, as determined by
list_image_pipeline_images	Returns a list of images created by the specified pipeline
list_image_pipelines	Returns a list of image pipelines
list_image_recipes	Returns a list of image recipes
list_images	Returns the list of images that you have access to
list_image_scan_finding_aggregations	Returns a list of image scan aggregations for your account
list_image_scan_findings	Returns a list of image scan findings for your account
list_infrastructure_configurations	Returns a list of infrastructure configurations
list_lifecycle_execution_resources	List resources that the runtime instance of the image lifecycle identified for lifecycle
list_lifecycle_executions	Get the lifecycle runtime history for the specified resource
list_lifecycle_policies	Get a list of lifecycle policies in your Amazon Web Services account

list_tags_for_resource	Returns the list of tags for the specified resource
list_waiting_workflow_steps	Get a list of workflow steps that are waiting for action for workflows in your Amazon
list_workflow_build_versions	Returns a list of build versions for a specific workflow resource
list_workflow_executions	Returns a list of workflow runtime instance metadata objects for a specific image bui
list_workflows	Lists workflow build versions based on filtering parameters
list_workflow_step_executions	Returns runtime data for each step in a runtime instance of the workflow that you spe
put_component_policy	Applies a policy to a component
put_container_recipe_policy	Applies a policy to a container image
put_image_policy	Applies a policy to an image
put_image_recipe_policy	Applies a policy to an image recipe
send_workflow_step_action	Pauses or resumes image creation when the associated workflow runs a WaitForActio
start_image_pipeline_execution	Manually triggers a pipeline to create an image
start_resource_state_update	Begin asynchronous resource state update for lifecycle changes to the specified imag
tag_resource	Adds a tag to a resource
untag_resource	Removes a tag from a resource
update_distribution_configuration	Updates a new distribution configuration
update_image_pipeline	Updates an image pipeline
update_infrastructure_configuration	Updates a new infrastructure configuration
update_lifecycle_policy	Update the specified lifecycle policy

Examples

```
## Not run:
svc <- imagebuilder()
svc$cancel_image_creation(
  Foo = 123
)

## End(Not run)
```

`lambda` *AWS Lambda*

Description

Lambda

Overview

Lambda is a compute service that lets you run code without provisioning or managing servers. Lambda runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, code monitoring and logging. With Lambda, you can run code for virtually any type of application or backend service. For more information about the Lambda service, see What is Lambda in the **Lambda Developer Guide**.

The *Lambda API Reference* provides information about each of the API methods, including details about the parameters in each API request and response.

You can use Software Development Kits (SDKs), Integrated Development Environment (IDE) Toolkits, and command line tools to access the API. For installation instructions, see Tools for Amazon Web Services.

For a list of Region-specific endpoints that Lambda supports, see Lambda endpoints and quotas in the *Amazon Web Services General Reference.*.

When making the API calls, you will need to authenticate your request by providing a signature. Lambda supports signature version 4. For more information, see Signature Version 4 signing process in the *Amazon Web Services General Reference.*.

CA certificates

Because Amazon Web Services SDKs use the CA certificates from your computer, changes to the certificates on the Amazon Web Services servers can cause connection failures when you attempt to use an SDK. You can prevent these failures by keeping your computer's CA certificates and operating system up-to-date. If you encounter this issue in a corporate environment and do not manage your own computer, you might need to ask an administrator to assist with the update process. The following list shows minimum operating system and Java versions:

- Microsoft Windows versions that have updates from January 2005 or later installed contain at least one of the required CAs in their trust list.
- Mac OS X 10.4 with Java for Mac OS X 10.4 Release 5 (February 2007), Mac OS X 10.5 (October 2007), and later versions contain at least one of the required CAs in their trust list.
- Red Hat Enterprise Linux 5 (March 2007), 6, and 7 and CentOS 5, 6, and 7 all contain at least one of the required CAs in their default trusted CA list.
- Java 1.4.2_12 (May 2006), 5 Update 2 (March 2005), and all later versions, including Java 6 (December 2006), 7, and 8, contain at least one of the required CAs in their default trusted CA list.

When accessing the Lambda management console or Lambda API endpoints, whether through browsers or programmatically, you will need to ensure your client machines support any of the following CAs:

- Amazon Root CA 1
- Starfield Services Root Certificate Authority - G2
- Starfield Class 2 Certification Authority

Root certificates from the first two authorities are available from Amazon trust services, but keeping your computer up-to-date is the more straightforward solution. To learn more about ACM-provided certificates, see Amazon Web Services Certificate Manager FAQs.

Usage

```
lambda(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)
```

Operations

add_layer_version_permission	Adds permissions to the resource-based policy of a version of an Lambda layer
add_permission	Grants a principal permission to use a function
create_alias	Creates an alias for a Lambda function version
create_code_signing_config	Creates a code signing configuration
create_event_source_mapping	Creates a mapping between an event source and an Lambda function
create_function	Creates a Lambda function
create_function_url_config	Creates a Lambda function URL with the specified configuration parameters
delete_alias	Deletes a Lambda function alias
delete_code_signing_config	Deletes the code signing configuration
delete_event_source_mapping	Deletes an event source mapping
delete_function	Deletes a Lambda function
delete_function_code_signing_config	Removes the code signing configuration from the function
delete_function_concurrency	Removes a concurrent execution limit from a function
delete_function_event_invoke_config	Deletes the configuration for asynchronous invocation for a function, version, or ali
delete_function_url_config	Deletes a Lambda function URL
delete_layer_version	Deletes a version of an Lambda layer
delete_provisioned_concurrency_config	Deletes the provisioned concurrency configuration for a function
get_account_settings	Retrieves details about your account's limits and usage in an Amazon Web Services
get_alias	Returns details about a Lambda function alias
get_code_signing_config	Returns information about the specified code signing configuration

get_event_source_mapping	Returns details about an event source mapping
get_function	Returns information about the function or function version, with a link to download
get_function_code_signing_config	Returns the code signing configuration for the specified function
get_function_concurrency	Returns details about the reserved concurrency configuration for a function
get_function_configuration	Returns the version-specific settings of a Lambda function or version
get_function_event_invoke_config	Retrieves the configuration for asynchronous invocation for a function, version, or a
get_function_recursion_config	Returns your function's recursive loop detection configuration
get_function_url_config	Returns details about a Lambda function URL
get_layer_version	Returns information about a version of an Lambda layer, with a link to download th
get_layer_version_by_arn	Returns information about a version of an Lambda layer, with a link to download th
get_layer_version_policy	Returns the permission policy for a version of an Lambda layer
get_policy	Returns the resource-based IAM policy for a function, version, or alias
get_provisioned_concurrency_config	Retrieves the provisioned concurrency configuration for a function's alias or versio
get_runtime_management_config	Retrieves the runtime management configuration for a function's version
invoke	Invokes a Lambda function
invoke_async	For asynchronous function invocation, use Invoke
invoke_with_response_stream	Configure your Lambda functions to stream response payloads back to clients
list_aliases	Returns a list of aliases for a Lambda function
list_code_signing_configs	Returns a list of code signing configurations
list_event_source_mappings	Lists event source mappings
list_function_event_invoke_configs	Retrieves a list of configurations for asynchronous invocation for a function
list_functions	Returns a list of Lambda functions, with the version-specific configuration of each
list_functions_by_code_signing_config	List the functions that use the specified code signing configuration
list_function_url_configs	Returns a list of Lambda function URLs for the specified function
list_layers	Lists Lambda layers and shows information about the latest version of each
list_layer_versions	Lists the versions of an Lambda layer
list_provisioned_concurrency_configs	Retrieves a list of provisioned concurrency configurations for a function
list_tags	Returns a function, event source mapping, or code signing configuration's tags
list_versions_by_function	Returns a list of versions, with the version-specific configuration of each
publish_layer_version	Creates an Lambda layer from a ZIP archive
publish_version	Creates a version from the current code and configuration of a function
put_function_code_signing_config	Update the code signing configuration for the function
put_function_concurrency	Sets the maximum number of simultaneous executions for a function, and reserves
put_function_event_invoke_config	Configures options for asynchronous invocation on a function, version, or alias
put_function_recursion_config	Sets your function's recursive loop detection configuration
put_provisioned_concurrency_config	Adds a provisioned concurrency configuration to a function's alias or version
put_runtime_management_config	Sets the runtime management configuration for a function's version
remove_layer_version_permission	Removes a statement from the permissions policy for a version of an Lambda layer
remove_permission	Revokes function-use permission from an Amazon Web Services service or another
tag_resource	Adds tags to a function, event source mapping, or code signing configuration
untag_resource	Removes tags from a function, event source mapping, or code signing configuration
update_alias	Updates the configuration of a Lambda function alias
update_code_signing_config	Update the code signing configuration
update_event_source_mapping	Updates an event source mapping
update_function_code	Updates a Lambda function's code
update_function_configuration	Modify the version-specific settings of a Lambda function
update_function_event_invoke_config	Updates the configuration for asynchronous invocation for a function, version, or al
update_function_url_config	Updates the configuration for a Lambda function URL

Examples

```
## Not run:
svc <- lambda()
# The following example grants permission for the account 223456789012 to
# use version 1 of a layer named my-layer.
svc$add_layer_version_permission(
  Action = "lambda:GetLayerVersion",
  LayerName = "my-layer",
  Principal = "223456789012",
  StatementId = "xaccount",
  VersionNumber = 1L
)

## End(Not run)
```

`lightsail` *Amazon Lightsail*

Description

Amazon Lightsail is the easiest way to get started with Amazon Web Services (Amazon Web Services) for developers who need to build websites or web applications. It includes everything you need to launch your project quickly - instances (virtual private servers), container services, storage buckets, managed databases, SSD-based block storage, static IP addresses, load balancers, content delivery network (CDN) distributions, DNS management of registered domains, and resource snapshots (backups) - for a low, predictable monthly price.

You can manage your Lightsail resources using the Lightsail console, Lightsail API, Command Line Interface (CLI), or SDKs. For more information about Lightsail concepts and tasks, see the Amazon Lightsail Developer Guide.

This API Reference provides detailed information about the actions, data types, parameters, and errors of the Lightsail service. For more information about the supported Amazon Web Services Regions, endpoints, and service quotas of the Lightsail service, see Amazon Lightsail Endpoints and Quotas in the *Amazon Web Services General Reference*.

Usage

```
lightsail(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)
```

Operations

allocate_static_ip	Allocates a static IP address
attach_certificate_to_distribution	Attaches an SSL/TLS certificate to your Amazon Lightsail content delivery
attach_disk	Attaches a block storage disk to a running or stopped Lightsail instance and
attach_instances_to_load_balancer	Attaches one or more Lightsail instances to a load balancer
attach_load_balancer_tls_certificate	Attaches a Transport Layer Security (TLS) certificate to your load balancer
attach_static_ip	Attaches a static IP address to a specific Amazon Lightsail instance
close_instance_public_ports	Closes ports for a specific Amazon Lightsail instance
copy_snapshot	Copies a manual snapshot of an instance or disk as another manual snapsho
create_bucket	Creates an Amazon Lightsail bucket
create_bucket_access_key	Creates a new access key for the specified Amazon Lightsail bucket
create_certificate	Creates an SSL/TLS certificate for an Amazon Lightsail content delivery ne
create_cloud_formation_stack	Creates an AWS CloudFormation stack, which creates a new Amazon EC2
create_contact_method	Creates an email or SMS text message contact method
create_container_service	Creates an Amazon Lightsail container service
create_container_service_deployment	Creates a deployment for your Amazon Lightsail container service
create_container_service_registry_login	Creates a temporary set of log in credentials that you can use to log in to the
create_disk	Creates a block storage disk that can be attached to an Amazon Lightsail ins
create_disk_from_snapshot	Creates a block storage disk from a manual or automatic snapshot of a disk
create_disk_snapshot	Creates a snapshot of a block storage disk
create_distribution	Creates an Amazon Lightsail content delivery network (CDN) distribution

create_domain	Creates a domain resource for the specified domain (example
create_domain_entry	Creates one of the following domain name system (DNS) records in a doma
create_gui_session_access_details	Creates two URLs that are used to access a virtual computer's graphical use
create_instances	Creates one or more Amazon Lightsail instances
create_instances_from_snapshot	Creates one or more new instances from a manual or automatic snapshot of
create_instance_snapshot	Creates a snapshot of a specific virtual private server, or instance
create_key_pair	Creates a custom SSH key pair that you can use with an Amazon Lightsail i
create_load_balancer	Creates a Lightsail load balancer
create_load_balancer_tls_certificate	Creates an SSL/TLS certificate for an Amazon Lightsail load balancer
create_relational_database	Creates a new database in Amazon Lightsail
create_relational_database_from_snapshot	Creates a new database from an existing database snapshot in Amazon Ligh
create_relational_database_snapshot	Creates a snapshot of your database in Amazon Lightsail
delete_alarm	Deletes an alarm
delete_auto_snapshot	Deletes an automatic snapshot of an instance or disk
delete_bucket	Deletes a Amazon Lightsail bucket
delete_bucket_access_key	Deletes an access key for the specified Amazon Lightsail bucket
delete_certificate	Deletes an SSL/TLS certificate for your Amazon Lightsail content delivery
delete_contact_method	Deletes a contact method
delete_container_image	Deletes a container image that is registered to your Amazon Lightsail conta
delete_container_service	Deletes your Amazon Lightsail container service
delete_disk	Deletes the specified block storage disk
delete_disk_snapshot	Deletes the specified disk snapshot
delete_distribution	Deletes your Amazon Lightsail content delivery network (CDN) distributio
delete_domain	Deletes the specified domain recordset and all of its domain records
delete_domain_entry	Deletes a specific domain entry
delete_instance	Deletes an Amazon Lightsail instance
delete_instance_snapshot	Deletes a specific snapshot of a virtual private server (or instance)
delete_key_pair	Deletes the specified key pair by removing the public key from Amazon Lig
delete_known_host_keys	Deletes the known host key or certificate used by the Amazon Lightsail bro
delete_load_balancer	Deletes a Lightsail load balancer and all its associated SSL/TLS certificates
delete_load_balancer_tls_certificate	Deletes an SSL/TLS certificate associated with a Lightsail load balancer
delete_relational_database	Deletes a database in Amazon Lightsail
delete_relational_database_snapshot	Deletes a database snapshot in Amazon Lightsail
detach_certificate_from_distribution	Detaches an SSL/TLS certificate from your Amazon Lightsail content deliv
detach_disk	Detaches a stopped block storage disk from a Lightsail instance
detach_instances_from_load_balancer	Detaches the specified instances from a Lightsail load balancer
detach_static_ip	Detaches a static IP from the Amazon Lightsail instance to which it is attacl
disable_add_on	Disables an add-on for an Amazon Lightsail resource
download_default_key_pair	Downloads the regional Amazon Lightsail default key pair
enable_add_on	Enables or modifies an add-on for an Amazon Lightsail resource
export_snapshot	Exports an Amazon Lightsail instance or block storage disk snapshot to Am
get_active_names	Returns the names of all active (not deleted) resources
get_alarms	Returns information about the configured alarms
get_auto_snapshots	Returns the available automatic snapshots for an instance or disk
get_blueprints	Returns the list of available instance images, or blueprints
get_bucket_access_keys	Returns the existing access key IDs for the specified Amazon Lightsail buck
get_bucket_bundles	Returns the bundles that you can apply to a Amazon Lightsail bucket
get_bucket_metric_data	Returns the data points of a specific metric for an Amazon Lightsail bucket

get_buckets	Returns information about one or more Amazon Lightsail buckets
get_bundles	Returns the bundles that you can apply to an Amazon Lightsail instance wh
get_certificates	Returns information about one or more Amazon Lightsail SSL/TLS certifica
get_cloud_formation_stack_records	Returns the CloudFormation stack record created as a result of the create cl
get_contact_methods	Returns information about the configured contact methods
get_container_api_metadata	Returns information about Amazon Lightsail containers, such as the current
get_container_images	Returns the container images that are registered to your Amazon Lightsail c
get_container_log	Returns the log events of a container of your Amazon Lightsail container se
get_container_service_deployments	Returns the deployments for your Amazon Lightsail container service
get_container_service_metric_data	Returns the data points of a specific metric of your Amazon Lightsail contai
get_container_service_powers	Returns the list of powers that can be specified for your Amazon Lightsail c
get_container_services	Returns information about one or more of your Amazon Lightsail container
get_cost_estimate	Retrieves information about the cost estimate for a specified resource
get_disk	Returns information about a specific block storage disk
get_disks	Returns information about all block storage disks in your AWS account and
get_disk_snapshot	Returns information about a specific block storage disk snapshot
get_disk_snapshots	Returns information about all block storage disk snapshots in your AWS acc
get_distribution_bundles	Returns the bundles that can be applied to your Amazon Lightsail content d
get_distribution_latest_cache_reset	Returns the timestamp and status of the last cache reset of a specific Amazo
get_distribution_metric_data	Returns the data points of a specific metric for an Amazon Lightsail content
get_distributions	Returns information about one or more of your Amazon Lightsail content d
get_domain	Returns information about a specific domain recordset
get_domains	Returns a list of all domains in the user's account
get_export_snapshot_records	Returns all export snapshot records created as a result of the export snapsho
get_instance	Returns information about a specific Amazon Lightsail instance, which is a
get_instance_access_details	Returns temporary SSH keys you can use to connect to a specific virtual pri
get_instance_metric_data	Returns the data points for the specified Amazon Lightsail instance metric,
get_instance_port_states	Returns the firewall port states for a specific Amazon Lightsail instance, the
get_instances	Returns information about all Amazon Lightsail virtual private servers, or in
get_instance_snapshot	Returns information about a specific instance snapshot
get_instance_snapshots	Returns all instance snapshots for the user's account
get_instance_state	Returns the state of a specific instance
get_key_pair	Returns information about a specific key pair
get_key_pairs	Returns information about all key pairs in the user's account
get_load_balancer	Returns information about the specified Lightsail load balancer
get_load_balancer_metric_data	Returns information about health metrics for your Lightsail load balancer
get_load_balancers	Returns information about all load balancers in an account
get_load_balancer_tls_certificates	Returns information about the TLS certificates that are associated with the s
get_load_balancer_tls_policies	Returns a list of TLS security policies that you can apply to Lightsail load b
get_operation	Returns information about a specific operation
get_operations	Returns information about all operations
get_operations_for_resource	Gets operations for a specific resource (an instance or a static IP)
get_regions	Returns a list of all valid regions for Amazon Lightsail
get_relational_database	Returns information about a specific database in Amazon Lightsail
get_relational_database_blueprints	Returns a list of available database blueprints in Amazon Lightsail
get_relational_database_bundles	Returns the list of bundles that are available in Amazon Lightsail
get_relational_database_events	Returns a list of events for a specific database in Amazon Lightsail
get_relational_database_log_events	Returns a list of log events for a database in Amazon Lightsail

get_relational_database_log_streams	Returns a list of available log streams for a specific database in Amazon Lig
get_relational_database_master_user_password	Returns the current, previous, or pending versions of the master user passwo
get_relational_database_metric_data	Returns the data points of the specified metric for a database in Amazon Lig
get_relational_database_parameters	Returns all of the runtime parameters offered by the underlying database so
get_relational_databases	Returns information about all of your databases in Amazon Lightsail
get_relational_database_snapshot	Returns information about a specific database snapshot in Amazon Lightsai
get_relational_database_snapshots	Returns information about all of your database snapshots in Amazon Lights
get_setup_history	Returns detailed information for five of the most recent SetupInstanceHttps
get_static_ip	Returns information about an Amazon Lightsail static IP
get_static_ips	Returns information about all static IPs in the user's account
import_key_pair	Imports a public SSH key from a specific key pair
is_vpc_peered	Returns a Boolean value indicating whether your Lightsail VPC is peered
open_instance_public_ports	Opens ports for a specific Amazon Lightsail instance, and specifies the IP ad
peer_vpc	Peers the Lightsail VPC with the user's default VPC
put_alarm	Creates or updates an alarm, and associates it with the specified metric
put_instance_public_ports	Opens ports for a specific Amazon Lightsail instance, and specifies the IP ad
reboot_instance	Restarts a specific instance
reboot_relational_database	Restarts a specific database in Amazon Lightsail
register_container_image	Registers a container image to your Amazon Lightsail container service
release_static_ip	Deletes a specific static IP from your account
reset_distribution_cache	Deletes currently cached content from your Amazon Lightsail content deliv
send_contact_method_verification	Sends a verification request to an email contact method to ensure it's owned
set_ip_address_type	Sets the IP address type for an Amazon Lightsail resource
set_resource_access_for_bucket	Sets the Amazon Lightsail resources that can access the specified Lightsail b
setup_instance_https	Creates an SSL/TLS certificate that secures traffic for your website
start_gui_session	Initiates a graphical user interface (GUI) session that's used to access a virtu
start_instance	Starts a specific Amazon Lightsail instance from a stopped state
start_relational_database	Starts a specific database from a stopped state in Amazon Lightsail
stop_gui_session	Terminates a web-based Amazon DCV session that's used to access a virtua
stop_instance	Stops a specific Amazon Lightsail instance that is currently running
stop_relational_database	Stops a specific database that is currently running in Amazon Lightsail
tag_resource	Adds one or more tags to the specified Amazon Lightsail resource
test_alarm	Tests an alarm by displaying a banner on the Amazon Lightsail console
unpeer_vpc	Unpeers the Lightsail VPC from the user's default VPC
untag_resource	Deletes the specified set of tag keys and their values from the specified Ama
update_bucket	Updates an existing Amazon Lightsail bucket
update_bucket_bundle	Updates the bundle, or storage plan, of an existing Amazon Lightsail bucket
update_container_service	Updates the configuration of your Amazon Lightsail container service, such
update_distribution	Updates an existing Amazon Lightsail content delivery network (CDN) dist
update_distribution_bundle	Updates the bundle of your Amazon Lightsail content delivery network (CD
update_domain_entry	Updates a domain recordset after it is created
update_instance_metadata_options	Modifies the Amazon Lightsail instance metadata parameters on a running c
update_load_balancer_attribute	Updates the specified attribute for a load balancer
update_relational_database	Allows the update of one or more attributes of a database in Amazon Lights
update_relational_database_parameters	Allows the update of one or more parameters of a database in Amazon Ligh

Examples

```
## Not run:
svc <- lightsail()
svc$allocate_static_ip(
  Foo = 123
)

## End(Not run)
```

`proton` *AWS Proton*

Description

This is the Proton Service API Reference. It provides descriptions, syntax and usage examples for each of the actions and data types for the Proton service.

The documentation for each action shows the Query API request parameters and the XML response.

Alternatively, you can use the Amazon Web Services CLI to access an API. For more information, see the Amazon Web Services Command Line Interface User Guide.

The Proton service is a two-pronged automation framework. Administrators create service templates to provide standardized infrastructure and deployment tooling for serverless and container based applications. Developers, in turn, select from the available service templates to automate their application or service deployments.

Because administrators define the infrastructure and tooling that Proton deploys and manages, they need permissions to use all of the listed API operations.

When developers select a specific infrastructure and tooling set, Proton deploys their applications. To monitor their applications that are running on Proton, developers need permissions to the service *create*, *list*, *update* and *delete* API operations and the service instance *list* and *update* API operations.

To learn more about Proton, see the Proton User Guide.

Ensuring Idempotency

When you make a mutating API request, the request typically returns a result before the asynchronous workflows of the operation are complete. Operations might also time out or encounter other server issues before they're complete, even if the request already returned a result. This might make it difficult to determine whether the request succeeded. Moreover, you might need to retry the request multiple times to ensure that the operation completes successfully. However, if the original request and the subsequent retries are successful, the operation occurs multiple times. This means that you might create more resources than you intended.

Idempotency ensures that an API request action completes no more than one time. With an idempotent request, if the original request action completes successfully, any subsequent retries complete successfully without performing any further actions. However, the result might contain updated information, such as the current creation status.

The following lists of APIs are grouped according to methods that ensure idempotency.

Idempotent create APIs with a client token

The API actions in this list support idempotency with the use of a *client token*. The corresponding Amazon Web Services CLI commands also support idempotency using a client token. A client token is a unique, case-sensitive string of up to 64 ASCII characters. To make an idempotent API request using one of these actions, specify a client token in the request. We recommend that you *don't* reuse the same client token for other API requests. If you don't provide a client token for these APIs, a default client token is automatically provided by SDKs.

Given a request action that has succeeded:

If you retry the request using the same client token and the same parameters, the retry succeeds without performing any further actions other than returning the original resource detail data in the response.

If you retry the request using the same client token, but one or more of the parameters are different, the retry throws a `ValidationException` with an `IdempotentParameterMismatch` error.

Client tokens expire eight hours after a request is made. If you retry the request with the expired token, a new resource is created.

If the original resource is deleted and you retry the request, a new resource is created.

Idempotent create APIs with a client token:

- CreateEnvironmentTemplateVersion
- CreateServiceTemplateVersion
- CreateEnvironmentAccountConnection

Idempotent create APIs

Given a request action that has succeeded:

If you retry the request with an API from this group, and the original resource *hasn't* been modified, the retry succeeds without performing any further actions other than returning the original resource detail data in the response.

If the original resource has been modified, the retry throws a `ConflictException`.

If you retry with different input parameters, the retry throws a `ValidationException` with an `IdempotentParameterMismatch` error.

Idempotent create APIs:

- CreateEnvironmentTemplate
- CreateServiceTemplate
- CreateEnvironment
- CreateService

Idempotent delete APIs

Given a request action that has succeeded:

When you retry the request with an API from this group and the resource was deleted, its metadata is returned in the response.

If you retry and the resource doesn't exist, the response is empty.

In both cases, the retry succeeds.

Idempotent delete APIs:

- DeleteEnvironmentTemplate
- DeleteEnvironmentTemplateVersion
- DeleteServiceTemplate
- DeleteServiceTemplateVersion
- DeleteEnvironmentAccountConnection

Asynchronous idempotent delete APIs

Given a request action that has succeeded:

If you retry the request with an API from this group, if the original request delete operation status is `DELETE_IN_PROGRESS`, the retry returns the resource detail data in the response without performing any further actions.

If the original request delete operation is complete, a retry returns an empty response.

Asynchronous idempotent delete APIs:

- DeleteEnvironment
- DeleteService

Usage

```
proton(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

- **creds**:

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

Operations

accept_environment_account_connection	In a management account, an environment account connection request is accep
cancel_component_deployment	Attempts to cancel a component deployment (for a component that is in the IN
cancel_environment_deployment	Attempts to cancel an environment deployment on an UpdateEnvironment acti
cancel_service_instance_deployment	Attempts to cancel a service instance deployment on an UpdateServiceInstance
cancel_service_pipeline_deployment	Attempts to cancel a service pipeline deployment on an UpdateServicePipeline
create_component	Create an Proton component
create_environment	Deploy a new environment
create_environment_account_connection	Create an environment account connection in an environment account so that e
create_environment_template	Create an environment template for Proton
create_environment_template_version	Create a new major or minor version of an environment template
create_repository	Create and register a link to a repository
create_service	Create an Proton service
create_service_instance	Create a service instance
create_service_sync_config	Create the Proton Ops configuration file
create_service_template	Create a service template
create_service_template_version	Create a new major or minor version of a service template
create_template_sync_config	Set up a template to create new template versions automatically by tracking a l
delete_component	Delete an Proton component resource
delete_deployment	Delete the deployment
delete_environment	Delete an environment
delete_environment_account_connection	In an environment account, delete an environment account connection
delete_environment_template	If no other major or minor versions of an environment template exist, delete th
delete_environment_template_version	If no other minor versions of an environment template exist, delete a major ver
delete_repository	De-register and unlink your repository
delete_service	Delete a service, with its instances and pipeline
delete_service_sync_config	Delete the Proton Ops file
delete_service_template	If no other major or minor versions of the service template exist, delete the ser
delete_service_template_version	If no other minor versions of a service template exist, delete a major version of
delete_template_sync_config	Delete a template sync configuration
get_account_settings	Get detail data for Proton account-wide settings
get_component	Get detailed data for a component
get_deployment	Get detailed data for a deployment
get_environment	Get detailed data for an environment
get_environment_account_connection	In an environment account, get the detailed data for an environment account co
get_environment_template	Get detailed data for an environment template
get_environment_template_version	Get detailed data for a major or minor version of an environment template
get_repository	Get detail data for a linked repository
get_repository_sync_status	Get the sync status of a repository used for Proton template sync
get_resources_summary	Get counts of Proton resources
get_service	Get detailed data for a service
get_service_instance	Get detailed data for a service instance
get_service_instance_sync_status	Get the status of the synced service instance
get_service_sync_blocker_summary	Get detailed data for the service sync blocker summary
get_service_sync_config	Get detailed information for the service sync configuration
get_service_template	Get detailed data for a service template
get_service_template_version	Get detailed data for a major or minor version of a service template

get_template_sync_config	Get detail data for a template sync configuration
get_template_sync_status	Get the status of a template sync
list_component_outputs	Get a list of component Infrastructure as Code (IaC) outputs
list_component_provisioned_resources	List provisioned resources for a component with details
list_components	List components with summary data
list_deployments	List deployments
list_environment_account_connections	View a list of environment account connections
list_environment_outputs	List the infrastructure as code outputs for your environment
list_environment_provisioned_resources	List the provisioned resources for your environment
list_environments	List environments with detail data summaries
list_environment_templates	List environment templates
list_environment_template_versions	List major or minor versions of an environment template with detail data
list_repositories	List linked repositories with detail data
list_repository_sync_definitions	List repository sync definitions with detail data
list_service_instance_outputs	Get a list service of instance Infrastructure as Code (IaC) outputs
list_service_instance_provisioned_resources	List provisioned resources for a service instance with details
list_service_instances	List service instances with summary data
list_service_pipeline_outputs	Get a list of service pipeline Infrastructure as Code (IaC) outputs
list_service_pipeline_provisioned_resources	List provisioned resources for a service and pipeline with details
list_services	List services with summaries of detail data
list_service_templates	List service templates with detail data
list_service_template_versions	List major or minor versions of a service template with detail data
list_tags_for_resource	List tags for a resource
notify_resource_deployment_status_change	Notify Proton of status changes to a provisioned resource when you use self-m
reject_environment_account_connection	In a management account, reject an environment account connection from anot
tag_resource	Tag a resource
untag_resource	Remove a customer tag from a resource
update_account_settings	Update Proton settings that are used for multiple services in the Amazon Web S
update_component	Update a component
update_environment	Update an environment
update_environment_account_connection	In an environment account, update an environment account connection to use a
update_environment_template	Update an environment template
update_environment_template_version	Update a major or minor version of an environment template
update_service	Edit a service description or use a spec to add and delete service instances
update_service_instance	Update a service instance
update_service_pipeline	Update the service pipeline
update_service_sync_blocker	Update the service sync blocker by resolving it
update_service_sync_config	Update the Proton Ops config file
update_service_template	Update a service template
update_service_template_version	Update a major or minor version of a service template
update_template_sync_config	Update template sync configuration parameters, except for the templateName a

Examples

```
## Not run:
svc <- proton()
svc$accept_environment_account_connection(
```

```
  Foo = 123
)

## End(Not run)
```

`serverlessapplicationrepository` *AWSServerlessApplicationRepository*

Description

The AWS Serverless Application Repository makes it easy for developers and enterprises to quickly find and deploy serverless applications in the AWS Cloud. For more information about serverless applications, see Serverless Computing and Applications on the AWS website.

The AWS Serverless Application Repository is deeply integrated with the AWS Lambda console, so that developers of all levels can get started with serverless computing without needing to learn anything new. You can use category keywords to browse for applications such as web and mobile backends, data processing applications, or chatbots. You can also search for applications by name, publisher, or event source. To use an application, you simply choose it, configure any required fields, and deploy it with a few clicks.

You can also easily publish applications, sharing them publicly with the community at large, or privately within your team or across your organization. To publish a serverless application (or app), you can use the AWS Management Console, AWS Command Line Interface (AWS CLI), or AWS SDKs to upload the code. Along with the code, you upload a simple manifest file, also known as the AWS Serverless Application Model (AWS SAM) template. For more information about AWS SAM, see AWS Serverless Application Model (AWS SAM) on the AWS Labs GitHub repository.

The AWS Serverless Application Repository Developer Guide contains more information about the two developer experiences available:

- Consuming Applications – Browse for applications and view information about them, including source code and readme files. Also install, configure, and deploy applications of your choosing.

  Publishing Applications – Configure and upload applications to make them available to other developers, and publish new versions of applications.

Usage

```
serverlessapplicationrepository(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

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

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

`credentials` Optional credentials shorthand for the config parameter

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

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

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

Service syntax

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

```
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)
```

Operations

create_application	Creates an application, optionally including an AWS SAM file to create the first applica
create_application_version	Creates an application version
create_cloud_formation_change_set	Creates an AWS CloudFormation change set for the given application
create_cloud_formation_template	Creates an AWS CloudFormation template
delete_application	Deletes the specified application
get_application	Gets the specified application
get_application_policy	Retrieves the policy for the application
get_cloud_formation_template	Gets the specified AWS CloudFormation template
list_application_dependencies	Retrieves the list of applications nested in the containing application
list_applications	Lists applications owned by the requester
list_application_versions	Lists versions for the specified application
put_application_policy	Sets the permission policy for an application
unshare_application	Unshares an application from an AWS Organization
update_application	Updates the specified application

Examples

```
## Not run:
```

```
svc <- serverlessapplicationrepository()
svc$create_application(
  Foo = 123
)

## End(Not run)
```

Index