

Package ‘smdocker’

June 7, 2023

Title Build 'Docker Images' in 'Amazon SageMaker Studio' using 'Amazon Web Service CodeBuild'

Version 0.1.4

Description

Allows users to easily build custom 'docker images' <<https://docs.docker.com/>> from 'Amazon Web Service Sagemaker' <<https://aws.amazon.com/sagemaker/>> using 'Amazon Web Service CodeBuild' <<https://aws.amazon.com/codebuild/>>.

License MIT + file LICENSE

URL <https://github.com/DyfanJones/sm-docker>

BugReports <https://github.com/DyfanJones/sm-docker/issues>

Encoding UTF-8

Imports jsonlite, paws.compute, paws.developer.tools, paws.machine.learning, paws.management, paws.storage, paws.security.identity, zip, stats, utils, uuid

Suggests covr, crayon, mockery, testthat (>= 3.0.0)

RoxygenNote 7.2.3

Collate 'utils.R' 'logs.R' 'code_build.R' 'builder.R' 'config.R' 'logging.R' 'sm_role.R' 'main.R' 'zzz.R'

Config/testthat/edition 3

NeedsCompilation no

Author Dyfan Jones [aut, cre]

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

Repository CRAN

Date/Publication 2023-06-07 09:50:02 UTC

R topics documented:

sagemaker_get_execution_role	2
smdocker_config	2
smdocker_log	3
sm_build	5

Index[7](#)

`sagemaker_get_execution_role`*Return the AWS ARN execution role from AWS SageMaker*

Description

Return the **AWS ARN execution role** from AWS SageMaker

Usage

```
sagemaker_get_execution_role()
```

Value

Character containing the AWS ARN role retrieved from AWS SageMaker

`smdocker_config`*Set paws config across smdocker package*

Description

This function sets up paws client config list for all AWS calls. This function only needs to be used when changing default settings when connecting to AWS.

Usage

```
smdocker_config(  
    aws_access_key_id = NULL,  
    aws_secret_access_key = NULL,  
    aws_session_token = NULL,  
    region_name = NULL,  
    profile_name = NULL,  
    disable_ssl = FALSE,  
    anonymous = FALSE,  
    refresh = FALSE,  
    ...  
)
```

Arguments

aws_access_key_id	(character): AWS access key ID
aws_secret_access_key	(character): AWS secret access key
aws_session_token	(character): AWS temporary session token
region_name	(character): Default region when creating new connections
profile_name	(character): The name of a profile to use. If not given, then the default profile is used.
disable_ssl	(logical): Whether or not to use SSL. By default, SSL is used.
anonymous	(logical): Set up anonymous credentials when connecting to AWS.
refresh	(logical): Refresh cached smdocker config
...	Other parameters within paws client.

Value

Invisible list, containing credentials for paws clients.

See Also

[ecr](#) [codebuild](#) [sagemaker](#) [cloudwatchlogs](#) [iam](#) [sts](#) [s3](#)

Examples

```
# Set up connection using profile
smdocker_config(profile_name = "smdocker_example")

# Reset connection to connect to a different region
smdocker_config(
  profile_name = "smdocker_example",
  region_name = "us-east-1",
  refresh = TRUE
)
```

smdocker_log

smdocker logging system

Description

Ability to configure smdocker logging system, through the use of smdocker helper function `smdocker_log` or `R:base` options function. options configurable parameters:

- `smdocker.log_level` (integer): The minimum log level that should be tracked
- `smdocker.log_file` (character): path for logs to populate, default output logs to console.
- `smdocker.log_timestamp_fmt` (character): see [format.POSIXct\(\)](#)

Usage

```
smdocker_log(  
  level = 3L,  
  file = "",  
  timestamp_fmt = "%Y-%m-%d %H:%M:%OS3"  
)
```

Arguments

level (integer): the level logging threshold.

- 4L : DEBUG
- 3L : INFO
- 2L : WARNING
- 1L : ERROR

file (character): path for logs to populate, default output logs to console.

timestamp_fmt (character): timestamp format, see [format.POSIXct\(\)](#).

Value

NULL invisible

Examples

```
## Not run:  
# log to a file  
temp_file <- tempfile()  
smdocker_log(file = temp_file)  
  
# change log threshold to INFO  
smdocker_log(level = 3L)  
  
# reset to default config  
smdocker_log()  
  
# options() equivalents:  
  
# log to a file  
temp_file <- tempfile()  
options(smdocker.log_file = temp_file)  
  
# change log threshold to INFO  
options(smdocker.log_level = 3L)  
  
## End(Not run)
```

sm_build	<i>Use AWS CodeBuild to build docker images and push them to Amazon ECR</i>
----------	---

Description

This function takes a directory containing a `dockerfile`, and builds it on [AWS CodeBuild](#). The resulting image is then stored in [AWS ECR](#) for later use.

Usage

```
sm_build(
  repository = NULL,
  compute_type = c("BUILD_GENERAL1_SMALL", "BUILD_GENERAL1_MEDIUM",
    "BUILD_GENERAL1_LARGE", "BUILD_GENERAL1_2XLARGE"),
  role = NULL,
  dir = ".",
  bucket = NULL,
  vpc_id = NULL,
  subnet_ids = list(),
  security_group_ids = list(),
  log = TRUE,
  ...
)
```

Arguments

repository	(character): The ECR repository:tag for the image (default: sagemaker-studio- <code>{domain_id}</code> :latest)
compute_type	(character): The CodeBuild compute type (default: BUILD_GENERAL1_SMALL)
role	(character): The IAM role name for CodeBuild to use (default: the Studio execution role).
dir	(character): Directory to build
bucket	(character): The S3 bucket to use for sending data to CodeBuild (if None, use the SageMaker SDK default bucket).
vpc_id	(character): The Id of the VPC that will host the CodeBuild Project (such as <code>vpc-05c09f91d48831c8c</code>).
subnet_ids	(list): List of subnet ids for the CodeBuild Project (such as <code>subnet-0b31f1863e9d31a67</code>)
security_group_ids	(list): List of security group ids for the CodeBuild Project (such as <code>sg-0ce4ec0d0414d2ddc</code>).
log	(logical): Show the logs of the running CodeBuild build
...	docker build parameters https://docs.docker.com/engine/reference/commandline/build/#options (NOTE: use "_" instead of "-" for example: docker optional parameter <code>build-arg</code> becomes <code>build_arg</code>)

Value

invisible character vector of AWS ECR image uri.

Examples

```
## Not run:
# Execute on current directory.
sm_build()

# Execute on different directory.
sm_build(dir = "my-project")

# Add extra docker arguments
sm_build(
  file = "/path/to/Dockerfile",
  build_arg = "foo=bar"
)

## End(Not run)
```

Index

cloudwatchlogs, [3](#)
codebuild, [3](#)

ecr, [3](#)

format.POSIXct(), [3](#), [4](#)

iam, [3](#)

s3, [3](#)
sagemaker, [3](#)
sagemaker_get_execution_role, [2](#)
sm_build, [5](#)
smdocker_config, [2](#)
smdocker_log, [3](#)
sts, [3](#)