

## » Data Source: `aws__acm__certificate`

Use this data source to get the ARN of a certificate in AWS Certificate Manager (ACM). The process of requesting and verifying a certificate in ACM requires some manual steps, which means that Terraform cannot automate the creation of ACM certificates. But using this data source, you can reference them by domain without having to hard code the ARNs as input.

### » Example Usage

```
data "aws_acm_certificate" "example" {
  domain      = "tf.example.com"
  statuses    = ["ISSUED"]
}
```

```
data "aws_acm_certificate" "example" {
  domain      = "tf.example.com"
  types       = ["AMAZON_ISSUED"]
  most_recent = true
}
```

### » Argument Reference

- **domain** - (Required) The domain of the certificate to look up. If no certificate is found with this name, an error will be returned.
- **statuses** - (Optional) A list of statuses on which to filter the returned list. Valid values are `PENDING_VALIDATION`, `ISSUED`, `INACTIVE`, `EXPIRED`, `VALIDATION_TIMED_OUT`, `REVOKED` and `FAILED`. If no value is specified, only certificates in the `ISSUED` state are returned.
- **types** - (Optional) A list of types on which to filter the returned list. Valid values are `AMAZON_ISSUED` and `IMPORTED`.
- **most\_recent** - (Optional) If set to true, it sorts the certificates matched by previous criteria by the `NotBefore` field, returning only the most recent one. If set to false, it returns an error if more than one certificate is found. Defaults to false.

### » Attributes Reference

- **arn** - Set to the ARN of the found certificate, suitable for referencing in other resources that support ACM certificates.

## » Data Source: `aws_lb`

**Note:** `aws_alb` is known as `aws_lb`. The functionality is identical.

Provides information about a Load Balancer.

This data source can prove useful when a module accepts an LB as an input variable and needs to, for example, determine the security groups associated with it, etc.

## » Example Usage

```
variable "lb_arn" {
  type    = "string"
  default = ""
}

variable "lb_name" {
  type    = "string"
  default = ""
}

data "aws_lb" "test" {
  arn    = "${var.lb_arn}"
  name   = "${var.lb_name}"
}
```

## » Argument Reference

The following arguments are supported:

- `arn` - (Optional) The full ARN of the load balancer.
- `name` - (Optional) The unique name of the load balancer.

**NOTE:** When both `arn` and `name` are specified, `arn` takes precedence.

## » Attributes Reference

See the LB Resource for details on the returned attributes - they are identical.

## » Data Source: `aws_lb_listener`

**Note:** `aws_alb_listener` is known as `aws_lb_listener`. The functionality is identical.

Provides information about a Load Balancer Listener.

This data source can prove useful when a module accepts an LB Listener as an input variable and needs to know the LB it is attached to, or other information specific to the listener in question.

## » Example Usage

```
# get listener from listener arn

variable "listener_arn" {
  type = "string"
}

data "aws_lb_listener" "listener" {
  arn = "${var.listener_arn}"
}

# get listener from load_balancer_arn and port

data "aws_lb" "selected" {
  name = "default-public"
}

data "aws_lb_listener" "selected443" {
  load_balancer_arn = "${data.aws_lb.selected.arn}"
  port = 443
}
```

## » Argument Reference

The following arguments are supported:

- `arn` - (Optional) The arn of the listener. Required if `load_balancer_arn` and `port` is not set.
- `load_balancer_arn` - (Optional) The arn of the load balancer. Required if `arn` is not set.
- `port` - (Optional) The port of the listener. Required if `arn` is not set.

## » Attributes Reference

See the LB Listener Resource for details on the returned attributes - they are identical.

## » Data Source: `aws_lb_target_group`

**Note:** `aws_alb_target_group` is known as `aws_lb_target_group`. The functionality is identical.

Provides information about a Load Balancer Target Group.

This data source can prove useful when a module accepts an LB Target Group as an input variable and needs to know its attributes. It can also be used to get the ARN of an LB Target Group for use in other resources, given LB Target Group name.

## » Example Usage

```
variable "lb_tg_arn" {
  type    = "string"
  default = ""
}

variable "lb_tg_name" {
  type    = "string"
  default = ""
}

data "aws_lb_target_group" "test" {
  arn    = "${var.lb_tg_arn}"
  name   = "${var.lb_tg_name}"
}
```

## » Argument Reference

The following arguments are supported:

- **arn** - (Optional) The full ARN of the target group.
- **name** - (Optional) The unique name of the target group.

**NOTE:** When both **arn** and **name** are specified, **arn** takes precedence.

## » Attributes Reference

See the LB Target Group Resource for details on the returned attributes - they are identical.

## » Data Source: `aws__ami`

Use this data source to get the ID of a registered AMI for use in other resources.

## » Example Usage

```
data "aws_ami" "nat_ami" {
  most_recent      = true
  executable_users = ["self"]

  filter {
    name   = "owner-alias"
    values = ["amazon"]
  }

  filter {
    name   = "name"
    values = ["amzn-ami-vpc-nat*"]
  }

  name_regex = "^myami-\\d{3}"
  owners     = ["self"]
}
```

## » Argument Reference

- `most_recent` - (Optional) If more than one result is returned, use the most recent AMI.
- `executable_users` - (Optional) Limit search to users with *explicit* launch permission on the image. Valid items are the numeric account ID or `self`.
- `filter` - (Optional) One or more name/value pairs to filter off of. There are several valid keys, for a full reference, check out `describe-images` in the AWS CLI reference.
- `owners` - (Optional) Limit search to specific AMI owners. Valid items are the numeric account ID, `amazon`, or `self`.

- **name\_regex** - (Optional) A regex string to apply to the AMI list returned by AWS. This allows more advanced filtering not supported from the AWS API. This filtering is done locally on what AWS returns, and could have a performance impact if the result is large. It is recommended to combine this with other options to narrow down the list AWS returns.

**NOTE:** At least one of **executable\_users**, **filter**, **owners**, or **name\_regex** must be specified.

**NOTE:** If more or less than a single match is returned by the search, Terraform will fail. Ensure that your search is specific enough to return a single AMI ID only, or use **most\_recent** to choose the most recent one. If you want to match multiple AMIs, use the **aws\_ami\_ids** data source instead.

## » Attributes Reference

**id** is set to the ID of the found AMI. In addition, the following attributes are exported:

**NOTE:** Some values are not always set and may not be available for interpolation.

- **architecture** - The OS architecture of the AMI (ie: i386 or x86\_64).
- **block\_device\_mappings** - The block device mappings of the AMI.
  - **block\_device\_mappings.#.device\_name** - The physical name of the device.
  - **block\_device\_mappings.#.ebs.delete\_on\_termination** - true if the EBS volume will be deleted on termination.
  - **block\_device\_mappings.#.ebs.encrypted** - true if the EBS volume is encrypted.
  - **block\_device\_mappings.#.ebs.iops** - 0 if the EBS volume is not a provisioned IOPS image, otherwise the supported IOPS count.
  - **block\_device\_mappings.#.ebs.snapshot\_id** - The ID of the snapshot.
  - **block\_device\_mappings.#.ebs.volume\_size** - The size of the volume, in GiB.
  - **block\_device\_mappings.#.ebs.volume\_type** - The volume type.
  - **block\_device\_mappings.#.no\_device** - Suppresses the specified device included in the block device mapping of the AMI.
  - **block\_device\_mappings.#.virtual\_name** - The virtual device name (for instance stores).
- **creation\_date** - The date and time the image was created.
- **description** - The description of the AMI that was provided during image creation.
- **hypervisor** - The hypervisor type of the image.
- **image\_id** - The ID of the AMI. Should be the same as the resource **id**.
- **image\_location** - The location of the AMI.

- `image_owner_alias` - The AWS account alias (for example, `amazon`, `self`) or the AWS account ID of the AMI owner.
- `image_type` - The type of image.
- `kernel_id` - The kernel associated with the image, if any. Only applicable for machine images.
- `name` - The name of the AMI that was provided during image creation.
- `owner_id` - The AWS account ID of the image owner.
- `platform` - The value is `Windows` for `Windows` AMIs; otherwise blank.
- `product_codes` - Any product codes associated with the AMI.
  - `product_codes.#.product_code_id` - The product code.
  - `product_codes.#.product_code_type` - The type of product code.
- `public` - `true` if the image has public launch permissions.
- `ramdisk_id` - The RAM disk associated with the image, if any. Only applicable for machine images.
- `root_device_name` - The device name of the root device.
- `root_device_type` - The type of root device (ie: `ebs` or `instance-store`).
- `root_snapshot_id` - The snapshot id associated with the root device, if any (only applies to `ebs` root devices).
- `sriov_net_support` - Specifies whether enhanced networking is enabled.
- `state` - The current state of the AMI. If the state is `available`, the image is successfully registered and can be used to launch an instance.
- `state_reason` - Describes a state change. Fields are `UNSET` if not available.
  - `state_reason.code` - The reason code for the state change.
  - `state_reason.message` - The message for the state change.
- `tags` - Any tags assigned to the image.
  - `tags.#.key` - The key name of the tag.
  - `tags.#.value` - The value of the tag.
- `virtualization_type` - The type of virtualization of the AMI (ie: `hvm` or `paravirtual`).

## » Data Source: `aws_ami_ids`

Use this data source to get a list of AMI IDs matching the specified criteria.

### » Example Usage

```
data "aws_ami_ids" "ubuntu" {
  owners = ["099720109477"]

  filter {
    name   = "name"
    values = ["ubuntu/images/ubuntu-*-*-amd64-server-*"]
  }
}
```

}

## » Argument Reference

- **executable\_users** - (Optional) Limit search to users with *explicit* launch permission on the image. Valid items are the numeric account ID or **self**.
- **filter** - (Optional) One or more name/value pairs to filter off of. There are several valid keys, for a full reference, check out describe-images in the AWS CLI reference.
- **owners** - (Optional) Limit search to specific AMI owners. Valid items are the numeric account ID, **amazon**, or **self**.
- **name\_regex** - (Optional) A regex string to apply to the AMI list returned by AWS. This allows more advanced filtering not supported from the AWS API. This filtering is done locally on what AWS returns, and could have a performance impact if the result is large. It is recommended to combine this with other options to narrow down the list AWS returns.

**NOTE:** At least one of **executable\_users**, **filter**, **owners** or **name\_regex** must be specified.

## » Attributes Reference

**ids** is set to the list of AMI IDs, sorted by creation time in descending order.

## » Data Source: `aws_api_gateway_rest_api`

Use this data source to get the `id` and `root_resource_id` of a REST API in API Gateway. To fetch the REST API you must provide a name to match against. As there is no unique name constraint on REST APIs this data source will error if there is more than one match.

## » Example Usage

```
data "aws_api_gateway_rest_api" "my_rest_api" {
  name = "my-rest-api"
}
```



## » Argument Reference

- **name** - (Required) The name of the REST API to look up. If no REST API is found with this name, an error will be returned. If multiple REST APIs are found with this name, an error will be returned.

## » Attributes Reference

- **id** - Set to the ID of the found REST API.
- **root\_resource\_id** - Set to the ID of the API Gateway Resource on the found REST API where the route matches '/'.

## » Data Source: `aws_autoscaling_groups`

The Autoscaling Groups data source allows access to the list of AWS ASGs within a specific region. This will allow you to pass a list of AutoScaling Groups to other resources.

## » Example Usage

```
data "aws_autoscaling_groups" "groups" {
  filter {
    name = "key"
    values = ["Team"]
  }

  filter {
    name = "value"
    values = ["Pets"]
  }
}

resource "aws_autoscaling_notification" "slack_notifications" {
  group_names = ["${data.aws_autoscaling_groups.groups.names}"]

  notifications = [
    "autoscaling:EC2_INSTANCE_LAUNCH",
    "autoscaling:EC2_INSTANCE_TERMINATE",
    "autoscaling:EC2_INSTANCE_LAUNCH_ERROR",
    "autoscaling:EC2_INSTANCE_TERMINATE_ERROR",
  ]
}
```

```

    topic_arn = "TOPIC ARN"
}

```

## » Argument Reference

- **filter** - (Optional) A filter used to scope the list e.g. by tags. See related docs.
  - **name** - (Required) The name of the filter. The valid values are: **auto-scaling-group**, **key**, **value**, and **propagate-at-launch**.
  - **values** - (Required) The value of the filter.

## » Attributes Reference

The following attributes are exported:

- **names** - A list of the Autoscaling Groups in the current region.

## » Data Source: `aws__availability__zone`

`aws__availability__zone` provides details about a specific availability zone (AZ) in the current region.

This can be used both to validate an availability zone given in a variable and to split the AZ name into its component parts of an AWS region and an AZ identifier letter. The latter may be useful e.g. for implementing a consistent subnet numbering scheme across several regions by mapping both the region and the subnet letter to network numbers.

This is different from the `aws__availability__zones` (plural) data source, which provides a list of the available zones.

## » Example Usage

The following example shows how this data source might be used to derive VPC and subnet CIDR prefixes systematically for an availability zone.

```

variable "region_number" {
  # Arbitrary mapping of region name to number to use in
  # a VPC's CIDR prefix.
  default = {
    us-east-1      = 1
    us-west-1      = 2
    us-west-2      = 3
  }
}

```

```

        eu-central-1    = 4
        ap-northeast-1 = 5
    }
}

variable "az_number" {
    # Assign a number to each AZ letter used in our configuration
    default = {
        a = 1
        b = 2
        c = 3
        d = 4
        e = 5
        f = 6
    }
}

# Retrieve the AZ where we want to create network resources
# This must be in the region selected on the AWS provider.
data "aws_availability_zone" "example" {
    name = "eu-central-1a"
}

# Create a VPC for the region associated with the AZ
resource "aws_vpc" "example" {
    cidr_block = "${cidrsubnet("10.0.0.0/8", 4, var.region_number[data.aws_availability_zone.
}

# Create a subnet for the AZ within the regional VPC
resource "aws_subnet" "example" {
    vpc_id      = "${aws_vpc.example.id}"
    cidr_block = "${cidrsubnet(aws_vpc.example.cidr_block, 4, var.az_number[data.aws_availabil
}

```

## » Argument Reference

The arguments of this data source act as filters for querying the available availability zones. The given filters must match exactly one availability zone whose data will be exported as attributes.

- **name** - (Optional) The full name of the availability zone to select.
- **state** - (Optional) A specific availability zone state to require. May be any of "available", "information" or "impaired".

All reasonable uses of this data source will specify **name**, since **state** alone would

match a single AZ only in a region that itself has only one AZ.

## » Attributes Reference

The following attributes are exported:

- **name** - The name of the selected availability zone.
- **region** - The region where the selected availability zone resides. This is always the region selected on the provider, since this data source searches only within that region.
- **name\_suffix** - The part of the AZ name that appears after the region name, uniquely identifying the AZ within its region.
- **state** - The current state of the AZ.

## » Data Source: `aws__availability__zones`

The Availability Zones data source allows access to the list of AWS Availability Zones which can be accessed by an AWS account within the region configured in the provider.

This is different from the `aws_availability_zone` (singular) data source, which provides some details about a specific availability zone.

## » Example Usage

```
# Declare the data source
data "aws_availability_zones" "available" {}

# e.g. Create subnets in the first two available availability zones

resource "aws_subnet" "primary" {
  availability_zone = "${data.aws_availability_zones.available.names[0]}"

  # ...
}

resource "aws_subnet" "secondary" {
  availability_zone = "${data.aws_availability_zones.available.names[1]}"

  # ...
}
```

## » Argument Reference

The following arguments are supported:

- **state** - (Optional) Allows to filter list of Availability Zones based on their current state. Can be either "available", "information", "impaired" or "unavailable". By default the list includes a complete set of Availability Zones to which the underlying AWS account has access, regardless of their state.

## » Attributes Reference

The following attributes are exported:

- **names** - A list of the Availability Zone names available to the account.

## » Data Source: `aws__batch__compute__environment`

The Batch Compute Environment data source allows access to details of a specific compute environment within AWS Batch.

## » Example Usage

```
data "aws_batch_compute_environment" "batch-mongo" {  
  compute_environment_name = "batch-mongo-production"  
}
```

## » Argument Reference

The following arguments are supported:

- **compute\_environment\_name** - (Required) The name of the Batch Compute Environment

## » Attributes Reference

The following attributes are exported:

- **arn** - The ARN of the compute environment.
- **ecs\_cluster\_arn** - The ARN of the underlying Amazon ECS cluster used by the compute environment.
- **service\_role** - The ARN of the IAM role that allows AWS Batch to make calls to other AWS services on your behalf.

- **type** - The type of the compute environment (for example, **MANAGED** or **UNMANAGED**).
- **status** - The current status of the compute environment (for example, **CREATING** or **VALID**).
- **status\_reason** - A short, human-readable string to provide additional details about the current status of the compute environment.
- **state** - The state of the compute environment (for example, **ENABLED** or **DISABLED**). If the state is **ENABLED**, then the compute environment accepts jobs from a queue and can scale out automatically based on queues.

## » Data Source: `aws_batch_job_queue`

The Batch Job Queue data source allows access to details of a specific job queue within AWS Batch.

### » Example Usage

```
data "aws_batch_job_queue" "test-queue" {
  name = "tf-test-batch-job-queue"
}
```

### » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the job queue.

### » Attributes Reference

The following attributes are exported:

- **arn** - The ARN of the job queue.
- **status** - The current status of the job queue (for example, **CREATING** or **VALID**).
- **status\_reason** - A short, human-readable string to provide additional details about the current status of the job queue.
- **state** - Describes the ability of the queue to accept new jobs (for example, **ENABLED** or **DISABLED**).
- **priority** - The priority of the job queue. Job queues with a higher priority are evaluated first when associated with the same compute environment.
- **compute\_environment\_order** - The compute environments that are attached to the job queue and the order in which job placement is preferred. Compute environments are selected for job placement in ascending order.

- `compute_environment_order.#.order` - The order of the compute environment.
- `compute_environment_order.#.compute_environment` - The ARN of the compute environment.

## » Data Source: `aws_billing_service_account`

Use this data source to get the Account ID of the AWS Billing and Cost Management Service Account for the purpose of whitelisting in S3 bucket policy.

### » Example Usage

```
data "aws_billing_service_account" "main" {}

resource "aws_s3_bucket" "billing_logs" {
  bucket = "my-billing-tf-test-bucket"
  acl    = "private"

  policy = <<POLICY
{
  "Id": "Policy",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetBucketAcl", "s3:GetBucketPolicy"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::my-billing-tf-test-bucket",
      "Principal": {
        "AWS": [
          "${data.aws_billing_service_account.main.arn}"
        ]
      }
    },
    {
      "Action": [
        "s3:PutObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::my-billing-tf-test-bucket/*",
      "Principal": {
        "AWS": [
```

```

        "${data.aws_billing_service_account.main.arn}"
    ]
}
}
]
}
POLICY
}

```

## » Attributes Reference

- `id` - The ID of the AWS billing service account.
- `arn` - The ARN of the AWS billing service account.

## » Data Source: `aws_caller_identity`

Use this data source to get the access to the effective Account ID, User ID, and ARN in which Terraform is authorized.

## » Example Usage

```

data "aws_caller_identity" "current" {}

output "account_id" {
  value = "${data.aws_caller_identity.current.account_id}"
}

output "caller_arn" {
  value = "${data.aws_caller_identity.current.arn}"
}

output "caller_user" {
  value = "${data.aws_caller_identity.current.user_id}"
}

```

## » Argument Reference

There are no arguments available for this data source.



## » Attributes Reference

- `account_id` - The AWS Account ID number of the account that owns or contains the calling entity.
- `arn` - The AWS ARN associated with the calling entity.
- `user_id` - The unique identifier of the calling entity.

## » Data Source: `aws__canonical__user__id`

The Canonical User ID data source allows access to the canonical user ID for the effective account in which Terraform is working.

## » Example Usage

```
data "aws__canonical__user__id" "current" {}

output "canonical_user_id" {
  value = "${data.aws__canonical__user__id.current.id}"
}
```

## » Argument Reference

There are no arguments available for this data source.

## » Attributes Reference

The following attributes are exported:

- `id` - The canonical user ID associated with the AWS account.
- `display_name` - The human-friendly name linked to the canonical user ID.

## » Data Source: `aws__cloudformation__stack`

The CloudFormation Stack data source allows access to stack outputs and other useful data including the template body.

## » Example Usage

```
data "aws_cloudformation_stack" "network" {
  name = "my-network-stack"
}

resource "aws_instance" "web" {
  ami           = "ami-abb07bcb"
  instance_type = "t1.micro"
  subnet_id     = "${data.aws_cloudformation_stack.network.outputs["SubnetId"]}"

  tags {
    Name = "HelloWorld"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the stack

## » Attributes Reference

The following attributes are exported:

- **capabilities** - A list of capabilities
- **description** - Description of the stack
- **disable\_rollback** - Whether the rollback of the stack is disabled when stack creation fails
- **notification\_arns** - A list of SNS topic ARNs to publish stack related events
- **outputs** - A map of outputs from the stack.
- **parameters** - A map of parameters that specify input parameters for the stack.
- **tags** - A map of tags associated with this stack.
- **template\_body** - Structure containing the template body.
- **iam\_role\_arn** - The ARN of the IAM role used to create the stack.
- **timeout\_in\_minutes** - The amount of time that can pass before the stack status becomes `CREATE_FAILED`

## » Data Source: `aws__cloudtrail__service__account`

Use this data source to get the Account ID of the AWS CloudTrail Service Account in a given region for the purpose of allowing CloudTrail to store trail data in S3.

## » Example Usage

```
data "aws_cloudtrail_service_account" "main" {}

resource "aws_s3_bucket" "bucket" {
  bucket          = "tf-cloudtrail-logging-test-bucket"
  force_destroy   = true

  policy = <<EOF
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "Put bucket policy needed for trails",
      "Effect": "Allow",
      "Principal": {
        "AWS": "${data.aws_cloudtrail_service_account.main.arn}"
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::tf-cloudtrail-logging-test-bucket/*"
    },
    {
      "Sid": "Get bucket policy needed for trails",
      "Effect": "Allow",
      "Principal": {
        "AWS": "${data.aws_cloudtrail_service_account.main.arn}"
      },
      "Action": "s3:GetBucketAcl",
      "Resource": "arn:aws:s3:::tf-cloudtrail-logging-test-bucket"
    }
  ]
}
EOF
}
```

## » Argument Reference

- **region** - (Optional) Name of the region whose AWS CloudTrail account ID is desired. Defaults to the region from the AWS provider configuration.

## » Attributes Reference

- **id** - The ID of the AWS CloudTrail service account in the selected region.
- **arn** - The ARN of the AWS CloudTrail service account in the selected region.

## » Data Source: `aws_cloudwatch_log_group`

Use this data source to get information about an AWS Cloudwatch Log Group

## » Example Usage

```
data "aws_cloudwatch_log_group" "example" {  
  name = "MyImportantLogs"  
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the Cloudwatch log group

## » Attributes Reference

The following attributes are exported:

- **arn** - The ARN of the Cloudwatch log group
- **creation\_time** - The creation time of the log group, expressed as the number of milliseconds after Jan 1, 1970 00:00:00 UTC.

## » Data Source: `aws_cognito_user_pools`

Use this data source to get a list of cognito user pools.

## » Example Usage

```
data "aws_api_gateway_rest_api" "selected" {
  name = "${var.api_gateway_name}"
}

data "aws_cognito_user_pools" "selected" {
  name = "${var.cognito_user_pool_name}"
}

resource "aws_api_gateway_authorizer" "cognito" {
  name = "cognito"
  type = "COGNITO_USER_POOLS"
  rest_api_id = "${data.aws_api_gateway_rest_api.selected.id}"
  provider_arns = ["${data.aws_cognito_user_pools.selected.arns}"]
}
```

## » Argument Reference

- **name** - (required) Name of the cognito user pools. Name is not a unique attribute for cognito user pool, so multiple pools might be returned with given name.

## » Attributes Reference

- **ids** - The list of cognito user pool ids.

## » Data Source: aws\_db\_instance

Use this data source to get information about an RDS instance

## » Example Usage

```
data "aws_db_instance" "database" {
  db_instance_identifier = "my-test-database"
}
```

## » Argument Reference

The following arguments are supported:

- **db\_instance\_identifier** - (Required) The name of the RDS instance

## » Attributes Reference

The following attributes are exported:

- **address** - The address of the RDS instance.
- **allocated\_storage** - Specifies the allocated storage size specified in gigabytes.
- **auto\_minor\_version\_upgrade** - Indicates that minor version patches are applied automatically.
- **availability\_zone** - Specifies the name of the Availability Zone the DB instance is located in.
- **backup\_retention\_period** - Specifies the number of days for which automatic DB snapshots are retained.
- **db\_cluster\_identifier** - If the DB instance is a member of a DB cluster, contains the name of the DB cluster that the DB instance is a member of.
- **db\_instance\_arn** - The Amazon Resource Name (ARN) for the DB instance.
- **db\_instance\_class** - Contains the name of the compute and memory capacity class of the DB instance.
- **db\_name** - Contains the name of the initial database of this instance that was provided at create time, if one was specified when the DB instance was created. This same name is returned for the life of the DB instance.
- **db\_parameter\_groups** - Provides the list of DB parameter groups applied to this DB instance.
- **db\_security\_groups** - Provides List of DB security groups associated to this DB instance.
- **db\_subnet\_group** - Specifies the name of the subnet group associated with the DB instance.
- **db\_instance\_port** - Specifies the port that the DB instance listens on.
- **endpoint** - The connection endpoint.
- **engine** - Provides the name of the database engine to be used for this DB instance.
- **engine\_version** - Indicates the database engine version.
- **hosted\_zone\_id** - The canonical hosted zone ID of the DB instance (to be used in a Route 53 Alias record).
- **iops** - Specifies the Provisioned IOPS (I/O operations per second) value.
- **kms\_key\_id** - If StorageEncrypted is true, the KMS key identifier for the encrypted DB instance.
- **license\_model** - License model information for this DB instance.
- **master\_username** - Contains the master username for the DB instance.
- **monitoring\_interval** - The interval, in seconds, between points when Enhanced Monitoring metrics are collected for the DB instance.
- **monitoring\_role\_arn** - The ARN for the IAM role that permits RDS to

send Enhanced Monitoring metrics to CloudWatch Logs.

- **multi\_az** - Specifies if the DB instance is a Multi-AZ deployment.
- **option\_group\_memberships** - Provides the list of option group memberships for this DB instance.
- **port** - The database port.
- **preferred\_backup\_window** - Specifies the daily time range during which automated backups are created.
- **preferred\_maintenance\_window** - Specifies the weekly time range during which system maintenance can occur in UTC.
- **publicly\_accessible** - Specifies the accessibility options for the DB instance.
- **storage\_encrypted** - Specifies whether the DB instance is encrypted.
- **storage\_type** - Specifies the storage type associated with DB instance.
- **timezone** - The time zone of the DB instance.
- **vpc\_security\_groups** - Provides a list of VPC security group elements that the DB instance belongs to.
- **replicate\_source\_db** - The identifier of the source DB that this is a replica of.
- **ca\_cert\_identifier** - Specifies the identifier of the CA certificate for the DB instance.

## » Data Source: `aws_db_snapshot`

Use this data source to get information about a DB Snapshot for use when provisioning DB instances

**NOTE:** This data source does not apply to snapshots created on Aurora DB clusters.

## » Example Usage

```
resource "aws_db_instance" "prod" {
  allocated_storage = 10
  engine            = "mysql"
  engine_version    = "5.6.17"
  instance_class    = "db.t2.micro"
  name              = "mydb"
  username          = "foo"
  password          = "bar"
  db_subnet_group_name = "my_database_subnet_group"
  parameter_group_name = "default.mysql5.6"
}

data "aws_db_snapshot" "latest_prod_snapshot" {
```

```

    db_instance_identifier = "${aws_db_instance.prod.id}"
    most_recent = true
}

# Use the latest production snapshot to create a dev instance.
resource "aws_db_instance" "dev" {
    instance_class      = "db.t2.micro"
    name                = "mydbdev"
    snapshot_identifier = "${data.aws_db_snapshot.latest_prod_snapshot.id}"
    lifecycle {
        ignore_changes = ["snapshot_identifier"]
    }
}

```

## » Argument Reference

The following arguments are supported:

- **most\_recent** - (Optional) If more than one result is returned, use the most recent Snapshot.
- **db\_instance\_identifier** - (Optional) Returns the list of snapshots created by the specific `db_instance`
- **db\_snapshot\_identifier** - (Optional) Returns information on a specific `snapshot_id`.
- **snapshot\_type** - (Optional) The type of snapshots to be returned. If you don't specify a `SnapshotType` value, then both automated and manual snapshots are returned. Shared and public DB snapshots are not included in the returned results by default. Possible values are, `automated`, `manual`, `shared` and `public`.
- **include\_shared** - (Optional) Set this value to `true` to include shared manual DB snapshots from other AWS accounts that this AWS account has been given permission to copy or restore, otherwise set this value to `false`. The default is `false`.
- **include\_public** - (Optional) Set this value to `true` to include manual DB snapshots that are public and can be copied or restored by any AWS account, otherwise set this value to `false`. The default is `false`.

## » Attributes Reference

The following attributes are exported:

- **id** - The snapshot ID.



- `allocated_storage` - Specifies the allocated storage size in gigabytes (GB).
- `availability_zone` - Specifies the name of the Availability Zone the DB instance was located in at the time of the DB snapshot.
- `db_snapshot_arn` - The Amazon Resource Name (ARN) for the DB snapshot.
- `encrypted` - Specifies whether the DB snapshot is encrypted.
- `engine` - Specifies the name of the database engine.
- `engine_version` - Specifies the version of the database engine.
- `iops` - Specifies the Provisioned IOPS (I/O operations per second) value of the DB instance at the time of the snapshot.
- `kms_key_id` - The ARN for the KMS encryption key.
- `license_model` - License model information for the restored DB instance.
- `option_group_name` - Provides the option group name for the DB snapshot.
- `source_db_snapshot_identifier` - The DB snapshot Arn that the DB snapshot was copied from. It only has value in case of cross customer or cross region copy.
- `source_region` - The region that the DB snapshot was created in or copied from.
- `status` - Specifies the status of this DB snapshot.
- `storage_type` - Specifies the storage type associated with DB snapshot.
- `vpc_id` - Specifies the storage type associated with DB snapshot.
- `snapshot_create_time` - Provides the time when the snapshot was taken, in Universal Coordinated Time (UTC).

## » Data Source: `aws_dynamodb_table`

Provides information about a DynamoDB table.

### » Example Usage

```
data "aws_dynamodb_table" "tableName" {
  name = "tableName"
}
```

### » Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the DynamoDB table.

## » Attributes Reference

See the DynamoDB Table Resource for details on the returned attributes - they are identical.

## » Data Source: `aws__ebs__snapshot`

Use this data source to get information about an EBS Snapshot for use when provisioning EBS Volumes

## » Example Usage

```
data "aws_ebs_snapshot" "ebs_volume" {
  most_recent = true
  owners      = ["self"]

  filter {
    name     = "volume-size"
    values   = ["40"]
  }

  filter {
    name     = "tag:Name"
    values   = ["Example"]
  }
}
```

## » Argument Reference

The following arguments are supported:

- `most_recent` - (Optional) If more than one result is returned, use the most recent snapshot.
- `owners` - (Optional) Returns the snapshots owned by the specified owner id. Multiple owners can be specified.
- `snapshot_ids` - (Optional) Returns information on a specific `snapshot_id`.
- `restorable_by_user_ids` - (Optional) One or more AWS accounts IDs that can create volumes from the snapshot.
- `filter` - (Optional) One or more name/value pairs to filter off of. There are several valid keys, for a full reference, check out `describe-snapshots` in the AWS CLI reference.

## » Attributes Reference

The following attributes are exported:

- `id` - The snapshot ID (e.g. `snap-59fcb34e`).
- `snapshot_id` - The snapshot ID (e.g. `snap-59fcb34e`).
- `description` - A description for the snapshot
- `owner_id` - The AWS account ID of the EBS snapshot owner.
- `owner_alias` - Value from an Amazon-maintained list (`amazon`, `aws-marketplace`, `microsoft`) of snapshot owners.
- `volume_id` - The volume ID (e.g. `vol-59fcb34e`).
- `encrypted` - Whether the snapshot is encrypted.
- `volume_size` - The size of the drive in GiBs.
- `kms_key_id` - The ARN for the KMS encryption key.
- `data_encryption_key_id` - The data encryption key identifier for the snapshot.
- `state` - The snapshot state.
- `tags` - A mapping of tags for the resource.

## » Data Source: `aws_ebs_snapshot_ids`

Use this data source to get a list of EBS Snapshot IDs matching the specified criteria.

## » Example Usage

```
data "aws_ebs_snapshot_ids" "ebs_volumes" {
  owners = ["self"]

  filter {
    name   = "volume-size"
    values = ["40"]
  }

  filter {
    name   = "tag:Name"
    values = ["Example"]
  }
}
```

## » Argument Reference

The following arguments are supported:

- **owners** - (Optional) Returns the snapshots owned by the specified owner id. Multiple owners can be specified.
- **restorable\_by\_user\_ids** - (Optional) One or more AWS accounts IDs that can create volumes from the snapshot.
- **filter** - (Optional) One or more name/value pairs to filter off of. There are several valid keys, for a full reference, check out describe-volumes in the AWS CLI reference.

## » Attributes Reference

**ids** is set to the list of EBS snapshot IDs, sorted by creation time in descending order.

## » Data Source: `aws_ebs_volume`

Use this data source to get information about an EBS volume for use in other resources.

## » Example Usage

```
data "aws_ebs_volume" "ebs_volume" {
  most_recent = true

  filter {
    name     = "volume-type"
    values = ["gp2"]
  }

  filter {
    name     = "tag:Name"
    values = ["Example"]
  }
}
```

## » Argument Reference

The following arguments are supported:

- **most\_recent** - (Optional) If more than one result is returned, use the most recent Volume.

- **filter** - (Optional) One or more name/value pairs to filter off of. There are several valid keys, for a full reference, check out describe-volumes in the AWS CLI reference.

## » Attributes Reference

The following attributes are exported:

- **id** - The volume ID (e.g. vol-59fcb34e).
- **volume\_id** - The volume ID (e.g. vol-59fcb34e).
- **arn** - The volume ARN (e.g. arn:aws:ec2:us-east-1:0123456789012:volume/vol-59fcb34e).
- **availability\_zone** - The AZ where the EBS volume exists.
- **encrypted** - Whether the disk is encrypted.
- **iops** - The amount of IOPS for the disk.
- **size** - The size of the drive in GiBs.
- **snapshot\_id** - The snapshot\_id the EBS volume is based off.
- **volume\_type** - The type of EBS volume.
- **kms\_key\_id** - The ARN for the KMS encryption key.
- **tags** - A mapping of tags for the resource.

## » Data Source: `aws__ecr__repository`

The ECR Repository data source allows the ARN, Repository URI and Registry ID to be retrieved for an ECR repository.

## » Example Usage

```
data "aws_ecr_repository" "service" {
  name = "ecr-repository"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the ECR Repository.

## » Attributes Reference

The following attributes are exported:

- `arn` - Full ARN of the repository.
- `registry_id` - The registry ID where the repository was created.
- `repository_url` - The URL of the repository (in the form `aws_account_id.dkr.ecr.region.amazonaws.com`).

## » Data Source: `aws__ecs__cluster`

The ECS Cluster data source allows access to details of a specific cluster within an AWS ECS service.

### » Example Usage

```
data "aws_ecs_cluster" "ecs-mongo" {
  cluster_name = "ecs-mongo-production"
}
```

### » Argument Reference

The following arguments are supported:

- `cluster_name` - (Required) The name of the ECS Cluster

### » Attributes Reference

The following attributes are exported:

- `arn` - The ARN of the ECS Cluster
- `status` - The status of the ECS Cluster
- `pending_tasks_count` - The number of pending tasks for the ECS Cluster
- `running_tasks_count` - The number of running tasks for the ECS Cluster
- `registered_container_instances_count` - The number of registered container instances for the ECS Cluster

## » Data Source: `aws__ecs__container__definition`

The ECS container definition data source allows access to details of a specific container within an AWS ECS service.

## » Example Usage

```
data "aws_ecs_container_definition" "ecs-mongo" {
  task_definition = "${aws_ecs_task_definition.mongo.id}"
  container_name  = "mongodb"
}
```

## » Argument Reference

The following arguments are supported:

- `task_definition` - (Required) The ARN of the task definition which contains the container
- `container_name` - (Required) The name of the container definition

## » Attributes Reference

The following attributes are exported:

- `image` - The docker image in use, including the digest
- `image_digest` - The digest of the docker image in use
- `cpu` - The CPU limit for this container definition
- `memory` - The memory limit for this container definition
- `memory_reservation` - The soft limit (in MiB) of memory to reserve for the container. When system memory is under contention, Docker attempts to keep the container memory to this soft limit
- `environment` - The environment in use
- `disable_networking` - Indicator if networking is disabled
- `docker_labels` - Set docker labels

## » Data Source: `aws__ecs__task__definition`

The ECS task definition data source allows access to details of a specific AWS ECS task definition.

## » Example Usage

```
# Simply specify the family to find the latest ACTIVE revision in that family.
data "aws_ecs_task_definition" "mongo" {
  task_definition = "${aws_ecs_task_definition.mongo.family}"
}
```

```

resource "aws_ecs_cluster" "foo" {
  name = "foo"
}

resource "aws_ecs_task_definition" "mongo" {
  family = "mongodb"

  container_definitions = <<DEFINITION
[
  {
    "cpu": 128,
    "environment": [{
      "name": "SECRET",
      "value": "KEY"
    }],
    "essential": true,
    "image": "mongo:latest",
    "memory": 128,
    "memoryReservation": 64,
    "name": "mongodb"
  }
]
DEFINITION
}

resource "aws_ecs_service" "mongo" {
  name           = "mongo"
  cluster        = "${aws_ecs_cluster.foo.id}"
  desired_count  = 2

  # Track the latest ACTIVE revision
  task_definition = "${aws_ecs_task_definition.mongo.family}:${max("${aws_ecs_task_definition
}

```

## » Argument Reference

The following arguments are supported:

- **task\_definition** - (Required) The family for the latest ACTIVE revision, family and revision (family:revision) for a specific revision in the family, the ARN of the task definition to access to.



## » Attributes Reference

The following attributes are exported:

- **family** - The family of this task definition
- **network\_mode** - The Docker networking mode to use for the containers in this task.
- **revision** - The revision of this task definition
- **status** - The status of this task definition
- **task\_role\_arn** - The ARN of the IAM role that containers in this task can assume

## » Data Source: `aws_efs_file_system`

Provides information about an Elastic File System (EFS).

## » Example Usage

```
variable "file_system_id" {
  type = "string"
  default = ""
}

data "aws_efs_file_system" "by_id" {
  file_system_id = "${var.file_system_id}"
}
```

## » Argument Reference

The following arguments are supported:

- **file\_system\_id** - (Optional) The ID that identifies the file system (e.g. fs-ccfc0d65).
- **creation\_token** - (Optional) Restricts the list to the file system with this creation token.

## » Attributes Reference

The following attributes are exported:

- **performance\_mode** - The PerformanceMode of the file system.
- **tags** - The list of tags assigned to the file system.
- **encrypted** - Whether EFS is encrypted.

- `kms_key_id` - The ARN for the KMS encryption key.
- `dns_name` - The DNS name for the filesystem per documented convention.

## » Data Source: `aws_efs_mount_target`

Provides information about an Elastic File System Mount Target (EFS).

### » Example Usage

```
variable "mount_target_id" {
  type = "string"
  default = ""
}

data "aws_efs_mount_target" "by_id" {
  mount_target_id = "${var.mount_target_id}"
}
```

### » Argument Reference

The following arguments are supported:

- `mount_target_id` - (Required) ID of the mount target that you want to have described

### » Attributes Reference

The following attributes are exported:

- `file_system_id` - ID of the file system for which the mount target is intended.
- `subnet_id` - ID of the mount target's subnet.
- `ip_address` - Address at which the file system may be mounted via the mount target.
- `security_groups` - List of VPC security group IDs attached to the mount target.
- `dns_name` - The DNS name for the given subnet/AZ per documented convention.
- `network_interface_id` - The ID of the network interface that Amazon EFS created when it created the mount target.

## » Data Source: `aws__eip`

`aws_eip` provides details about a specific Elastic IP.

This resource can prove useful when a module accepts an allocation ID or public IP as an input variable and needs to determine the other.

## » Example Usage

The following example shows how one might accept a public IP as a variable and use this data source to obtain the allocation ID.

```
variable "instance_id" {}
variable "public_ip" {}

data "aws_eip" "proxy_ip" {
  public_ip = "${var.public_ip}"
}

resource "aws_eip_association" "proxy_eip" {
  instance_id   = "${var.instance_id}"
  allocation_id = "${data.aws_eip.proxy_ip.id}"
}
```

## » Argument Reference

The arguments of this data source act as filters for querying the available Elastic IPs in the current region. The given filters must match exactly one Elastic IP whose data will be exported as attributes.

- `id` - (Optional) The allocation id of the specific EIP to retrieve.
- `public_ip` - (Optional) The public IP of the specific EIP to retrieve.

## » Attributes Reference

All of the argument attributes are also exported as result attributes. This data source will complete the data by populating any fields that are not included in the configuration with the data for the selected Elastic IP.

## » Data Source: `aws__elastic__beanstalk__hosted__zone`

Use this data source to get the ID of an elastic beanstalk hosted zone.

## » Example Usage

```
data "aws_elastic_beanstalk_hosted_zone" "current" {}
```

## » Argument Reference

- **region** - (Optional) The region you'd like the zone for. By default, fetches the current region.

## » Attributes Reference

- **id** - The ID of the hosted zone.
- **region** - The region of the hosted zone.

## » Data Source: `aws__elastic__beanstalk__solution__stack`

Use this data source to get the name of a elastic beanstalk solution stack.

## » Example Usage

```
data "aws_elastic_beanstalk_solution_stack" "multi_docker" {
  most_recent = true

  name_regex = "^64bit Amazon Linux (.*?) Multi-container Docker (.*?)$"
}
```

## » Argument Reference

- **most\_recent** - (Optional) If more than one result is returned, use the most recent solution stack.
- **name\_regex** - A regex string to apply to the solution stack list returned by AWS. See Elastic Beanstalk Supported Platforms from AWS documentation for reference solution stack names.

**NOTE:** If more or less than a single match is returned by the search, Terraform will fail. Ensure that your search is specific enough to return a single solution stack, or use **most\_recent** to choose the most recent one.

## » Attributes Reference

- `name` - The name of the solution stack.

## » Data Source: `aws__elasticache__cluster`

Use this data source to get information about an ElastiCache Cluster

## » Example Usage

```
data "aws_elasticache_cluster" "my_cluster" {
  cluster_id = "my-cluster-id"
}
```

## » Argument Reference

The following arguments are supported:

- `cluster_id` - (Required) Group identifier.

## » Attributes Reference

The following attributes are exported:

- `node_type` - The cluster node type.
- `num_cache_nodes` - The number of cache nodes that the cache cluster has.
- `engine` - Name of the cache engine.
- `engine_version` - Version number of the cache engine.
- `subnet_group_name` - Name of the subnet group associated to the cache cluster.
- `security_group_names` - List of security group names associated with this cache cluster.
- `security_group_ids` - List VPC security groups associated with the cache cluster.
- `parameter_group_name` - Name of the parameter group associated with this cache cluster.
- `replication_group_id` - The replication group to which this cache cluster belongs.
- `maintenance_window` - Specifies the weekly time range for when maintenance on the cache cluster is performed.

- `snapshot_window` - The daily time range (in UTC) during which ElastiCache will begin taking a daily snapshot of the cache cluster.
- `snapshot_retention_limit` - The number of days for which ElastiCache will retain automatic cache cluster snapshots before deleting them.
- `availability_zone` - The Availability Zone for the cache cluster.
- `notification_topic_arn` - An Amazon Resource Name (ARN) of an SNS topic that ElastiCache notifications get sent to.
- `port` - The port number on which each of the cache nodes will accept connections.
- `configuration_endpoint` - The configuration endpoint to allow host discovery.
- `cluster_address` - The DNS name of the cache cluster without the port appended.
- `cache_nodes` - List of node objects including `id`, `address`, `port` and `availability_zone`. Referenceable e.g. as `data.aws_elasticache_cluster.bar.cache_nodes.0.ad`
- `tags` - The tags assigned to the resource

## » Data Source: `aws_elasticache_replication_group`

Use this data source to get information about an ElastiCache Replication Group.

### » Example Usage

```
data "aws_elasticache_replication_group" "bar" {
  replication_group_id = "example"
}
```

### » Argument Reference

The following arguments are supported:

- `replication_group_id` - (Required) The identifier for the replication group.

### » Attributes Reference

The following attributes are exported:

- `replication_group_id` - The identifier for the replication group.
- `replication_group_description` - The description of the replication group.

- **auth\_token\_enabled** - A flag that enables using an AuthToken (password) when issuing Redis commands.
- **automatic\_failover\_enabled** - A flag whether a read-only replica will be automatically promoted to read/write primary if the existing primary fails.
- **node\_type** - The cluster node type.
- **number\_cache\_clusters** - The number of cache clusters that the replication group has.
- **snapshot\_window** - The daily time range (in UTC) during which ElastiCache begins taking a daily snapshot of your node group (shard).
- **snapshot\_retention\_limit** - The number of days for which ElastiCache retains automatic cache cluster snapshots before deleting them.
- **port** - The port number on which the configuration endpoint will accept connections.
- **configuration\_endpoint\_address** - The configuration endpoint address to allow host discovery.
- **primary\_endpoint\_address** - The endpoint of the primary node in this node group (shard).

## » **aws\_elb**

Provides information about a "classic" Elastic Load Balancer (ELB). See LB Data Source if you are looking for "v2" Application Load Balancer (ALB) or Network Load Balancer (NLB).

This data source can prove useful when a module accepts an LB as an input variable and needs to, for example, determine the security groups associated with it, etc.

## » **Example Usage**

```
variable "lb_name" {
  type      = "string"
  default   = ""
}

data "aws_elb" "test" {
  name = "${var.lb_name}"
}
```

## » **Argument Reference**

The following arguments are supported:

- **name** - (Required) The unique name of the load balancer.

## » Attributes Reference

See the ELB Resource for details on the returned attributes - they are identical.

## » Data Source: `aws_elb_hosted_zone_id`

Use this data source to get the HostedZoneId of the AWS Elastic Load Balancing HostedZoneId in a given region for the purpose of using in an AWS Route53 Alias.

## » Example Usage

```
data "aws_elb_hosted_zone_id" "main" {}

resource "aws_route53_record" "www" {
  zone_id = "${aws_route53_zone.primary.zone_id}"
  name     = "example.com"
  type     = "A"

  alias {
    name          = "${aws_elb.main.dns_name}"
    zone_id       = "${data.aws_elb_hosted_zone_id.main.id}"
    evaluate_target_health = true
  }
}
```

## » Argument Reference

- **region** - (Optional) Name of the region whose AWS ELB HostedZoneId is desired. Defaults to the region from the AWS provider configuration.

## » Attributes Reference

- **id** - The ID of the AWS ELB HostedZoneId in the selected region.



## » Data Source: `aws_elb_service_account`

Use this data source to get the Account ID of the AWS Elastic Load Balancing Service Account in a given region for the purpose of whitelisting in S3 bucket policy.

## » Example Usage

```
data "aws_elb_service_account" "main" {}

resource "aws_s3_bucket" "elb_logs" {
  bucket = "my-elb-tf-test-bucket"
  acl    = "private"

  policy = <<POLICY
{
  "Id": "Policy",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:PutObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::my-elb-tf-test-bucket/AWSLogs/*",
      "Principal": {
        "AWS": [
          "${data.aws_elb_service_account.main.arn}"
        ]
      }
    }
  ]
}
POLICY
}

resource "aws_elb" "bar" {
  name                = "my-foobar-terraform-elb"
  availability_zones = ["us-west-2a"]

  access_logs {
    bucket = "${aws_s3_bucket.elb_logs.bucket}"
    interval = 5
  }
}
```

```

listener {
  instance_port      = 8000
  instance_protocol = "http"
  lb_port           = 80
  lb_protocol       = "http"
}
}

```

## » Argument Reference

- **region** - (Optional) Name of the region whose AWS ELB account ID is desired. Defaults to the region from the AWS provider configuration.

## » Attributes Reference

- **id** - The ID of the AWS ELB service account in the selected region.
- **arn** - The ARN of the AWS ELB service account in the selected region.

## » Data Source: `aws_iam_account_alias`

The IAM Account Alias data source allows access to the account alias for the effective account in which Terraform is working.

## » Example Usage

```

data "aws_iam_account_alias" "current" {}

output "account_id" {
  value = "${data.aws_iam_account_alias.current.account_alias}"
}

```

## » Argument Reference

There are no arguments available for this data source.

## » Attributes Reference

The following attributes are exported:

- `account_alias` - The alias associated with the AWS account.

## » Data Source: `aws_iam_group`

This data source can be used to fetch information about a specific IAM group. By using this data source, you can reference IAM group properties without having to hard code ARNs as input.

### » Example Usage

```
data "aws_iam_group" "example" {
  group_name = "an_example_group_name"
}
```

### » Argument Reference

- `group_name` - (Required) The friendly IAM group name to match.

### » Attributes Reference

- `arn` - The Amazon Resource Name (ARN) specifying the group.
- `path` - The path to the group.
- `group_id` - The stable and unique string identifying the group.

## » Data Source: `aws_iam_instance_profile`

This data source can be used to fetch information about a specific IAM instance profile. By using this data source, you can reference IAM instance profile properties without having to hard code ARNs as input.

### » Example Usage

```
data "aws_iam_instance_profile" "example" {
  name = "an_example_instance_profile_name"
}
```

## » Argument Reference

- **name** - (Required) The friendly IAM instance profile name to match.

## » Attributes Reference

- **arn** - The Amazon Resource Name (ARN) specifying the instance profile.
- **create\_date** - The string representation of the date the instance profile was created.
- **path** - The path to the instance profile.
- **role\_arn** - The role arn associated with this instance profile.
- **role\_id** - The role id associated with this instance profile.
- **role\_name** - The role name associated with this instance profile.

## » aws\_iam\_policy

This data source can be used to fetch information about a specific IAM policy.

## » Example Usage

```
data "aws_iam_policy" "example" {  
  arn = "arn:aws:iam::123456789012:policy/UsersManageOwnCredentials"  
}
```

## » Argument Reference

- **arn** - (Required) ARN of the IAM policy.

## » Attributes Reference

- **name** - The name of the IAM policy.
- **arn** - The Amazon Resource Name (ARN) specifying the policy.
- **path** - The path to the policy.
- **description** - The description of the policy.
- **policy** - The policy document of the policy.

## » Data Source: `aws_iam_policy_document`

Generates an IAM policy document in JSON format.

This is a data source which can be used to construct a JSON representation of an IAM policy document, for use with resources which expect policy documents, such as the `aws_iam_policy` resource.

```
data "aws_iam_policy_document" "example" {
  statement {
    sid = "1"

    actions = [
      "s3:ListAllMyBuckets",
      "s3:GetBucketLocation",
    ]

    resources = [
      "arn:aws:s3:::*"
    ]
  }

  statement {
    actions = [
      "s3:ListBucket",
    ]

    resources = [
      "arn:aws:s3:::${var.s3_bucket_name}",
    ]

    condition {
      test      = "StringLike"
      variable = "s3:prefix"

      values = [
        "",
        "home/",
        "home/{aws:username}/",
      ]
    }
  }

  statement {
    actions = [
      "s3:*"
    ]
  }
}
```

```

    ]

    resources = [
        "arn:aws:s3:::${var.s3_bucket_name}/home/&{aws:username}",
        "arn:aws:s3:::${var.s3_bucket_name}/home/&{aws:username}/*",
    ]
}

resource "aws_iam_policy" "example" {
    name     = "example_policy"
    path     = "/"
    policy   = "${data.aws_iam_policy_document.example.json}"
}

```

Using this data source to generate policy documents is *optional*. It is also valid to use literal JSON strings within your configuration, or to use the `file` interpolation function to read a raw JSON policy document from a file.

## » Argument Reference

The following arguments are supported:

- **policy\_id** (Optional) - An ID for the policy document.
- **source\_json** (Optional) - An IAM policy document to import as a base for the current policy document. Statements with non-blank `sids` in the current policy document will overwrite statements with the same `sid` in the source json. Statements without an `sid` cannot be overwritten.
- **override\_json** (Optional) - An IAM policy document to import and override the current policy document. Statements with non-blank `sids` in the override document will overwrite statements with the same `sid` in the current document. Statements without an `sid` cannot be overwritten.
- **statement** (Required) - A nested configuration block (described below) configuring one *statement* to be included in the policy document.

Each document configuration must have one or more **statement** blocks, which each accept the following arguments:

- **sid** (Optional) - An ID for the policy statement.
- **effect** (Optional) - Either "Allow" or "Deny", to specify whether this statement allows or denies the given actions. The default is "Allow".
- **actions** (Optional) - A list of actions that this statement either allows or denies. For example, ["ec2:RunInstances", "s3:\*"].
- **not\_actions** (Optional) - A list of actions that this statement does *not* apply to. Used to apply a policy statement to all actions *except* those listed.

- **resources** (Optional) - A list of resource ARNs that this statement applies to. This is required by AWS if used for an IAM policy.
- **not\_resources** (Optional) - A list of resource ARNs that this statement does *not* apply to. Used to apply a policy statement to all resources *except* those listed.
- **principals** (Optional) - A nested configuration block (described below) specifying a resource (or resource pattern) to which this statement applies.
- **not\_principals** (Optional) - Like **principals** except gives resources that the statement does *not* apply to.
- **condition** (Optional) - A nested configuration block (described below) that defines a further, possibly-service-specific condition that constrains whether this statement applies.

Each policy may have either zero or more **principals** blocks or zero or more **not\_principals** blocks, both of which each accept the following arguments:

- **type** (Required) The type of principal. For AWS accounts this is "AWS".
- **identifiers** (Required) List of identifiers for principals. When **type** is "AWS", these are IAM user or role ARNs.

Each policy statement may have zero or more **condition** blocks, which each accept the following arguments:

- **test** (Required) The name of the IAM condition type to evaluate.
- **variable** (Required) The name of a Context Variable to apply the condition to. Context variables may either be standard AWS variables starting with **aws:**, or service-specific variables prefixed with the service name.
- **values** (Required) The values to evaluate the condition against. If multiple values are provided, the condition matches if at least one of them applies. (That is, the tests are combined with the "OR" boolean operation.)

When multiple **condition** blocks are provided, they must *all* evaluate to true for the policy statement to apply. (In other words, the conditions are combined with the "AND" boolean operation.)

## » Context Variable Interpolation

The IAM policy document format allows context variables to be interpolated into various strings within a statement. The native IAM policy document format uses `${...}`-style syntax that is in conflict with Terraform's interpolation syntax, so this data source instead uses `&{...}` syntax for interpolations that should be processed by AWS rather than by Terraform.

## » Attributes Reference

The following attribute is exported:

- `json` - The above arguments serialized as a standard JSON policy document.

## » Example with Multiple Principals

Showing how you can use this as an assume role policy as well as showing how you can specify multiple principal blocks with different types.

```
data "aws_iam_policy_document" "event_stream_bucket_role_assume_role_policy" {
  statement {
    actions = ["sts:AssumeRole"]

    principals {
      type       = "Service"
      identifiers = ["firehose.amazonaws.com"]
    }

    principals {
      type       = "AWS"
      identifiers = ["${var.trusted_role_arn}"]
    }
  }
}
```

## » Example with Source and Override

Showing how you can use `source_json` and `override_json`

```
data "aws_iam_policy_document" "source" {
  statement {
    actions    = ["ec2:*"]
    resources  = ["*"]
  }

  statement {
    sid = "SidToOverwrite"

    actions    = ["s3:*"]
    resources  = ["*"]
  }
}
```



```

data "aws_iam_policy_document" "source_json_example" {
  source_json = "${data.aws_iam_policy_document.source.json}"

  statement {
    sid = "SidToOverwrite"

    actions = ["s3:*"]

    resources = [
      "arn:aws:s3:::somebucket",
      "arn:aws:s3:::somebucket/*",
    ]
  }
}

data "aws_iam_policy_document" "override" {
  statement {
    sid = "SidToOverwrite"

    actions   = ["s3:*"]
    resources = ["*"]
  }
}

data "aws_iam_policy_document" "override_json_example" {
  override_json = "${data.aws_iam_policy_document.override.json}"

  statement {
    actions   = ["ec2:*"]
    resources = ["*"]
  }

  statement {
    sid = "SidToOverwrite"

    actions = ["s3:*"]

    resources = [
      "arn:aws:s3:::somebucket",
      "arn:aws:s3:::somebucket/*",
    ]
  }
}

data.aws_iam_policy_document.source_json_example.json will evaluate

```

to:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Action": "ec2:*",
      "Resource": "*"
    },
    {
      "Sid": "SidToOverwrite",
      "Effect": "Allow",
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::somebucket/*",
        "arn:aws:s3:::somebucket"
      ]
    }
  ]
}
```

`data.aws_iam_policy_document.override_json_example.json` will evaluate to:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Action": "ec2:*",
      "Resource": "*"
    },
    {
      "Sid": "SidToOverwrite",
      "Effect": "Allow",
      "Action": "s3:*",
      "Resource": "*"
    }
  ]
}
```

You can also combine `source_json` and `override_json` in the same document.

## » Data Source: `aws_iam_role`

This data source can be used to fetch information about a specific IAM role. By using this data source, you can reference IAM role properties without having to hard code ARNs as input.

### » Example Usage

```
data "aws_iam_role" "example" {
  name = "an_example_role_name"
}
```

### » Argument Reference

- `name` - (Required) The friendly IAM role name to match.

### » Attributes Reference

- `id` - The friendly IAM role name to match.
- `arn` - The Amazon Resource Name (ARN) specifying the role.
- `assume_role_policy` - The policy document associated with the role.
- `path` - The path to the role.
- `unique_id` - The stable and unique string identifying the role.

## » Data Source: `aws_iam_server_certificate`

Use this data source to lookup information about IAM Server Certificates.

### » Example Usage

```
data "aws_iam_server_certificate" "my-domain" {
  name_prefix = "my-domain.org"
  latest      = true
}
```

```
resource "aws_elb" "elb" {
  name = "my-domain-elb"

  listener {
    instance_port = 8000
  }
}
```

```

    instance_protocol = "https"
    lb_port           = 443
    lb_protocol       = "https"
    ssl_certificate_id = "${data.aws_iam_server_certificate.my-domain.arn}"
  }
}

```

## » Argument Reference

- `name_prefix` - prefix of cert to filter by
- `path_prefix` - prefix of path to filter by
- `name` - exact name of the cert to lookup
- `latest` - sort results by expiration date. returns the certificate with expiration date in furthest in the future.

## » Attributes Reference

- `arn` is set to the ARN of the IAM Server Certificate
- `path` is set to the path of the IAM Server Certificate
- `expiration_date` is set to the expiration date of the IAM Server Certificate
- `upload_date` is the date when the server certificate was uploaded
- `certificate_body` is the public key certificate (PEM-encoded). This is useful when configuring back-end instance authentication policy for load balancer
- `certificate_chain` is the public key certificate chain (PEM-encoded) if exists, empty otherwise

## » Import

The terraform import function will read in certificate body, certificate chain (if it exists), id, name, path, and arn. It will not retrieve the private key which is not available through the AWS API.

## » Data Source: `aws_iam_user`

This data source can be used to fetch information about a specific IAM user. By using this data source, you can reference IAM user properties without having to hard code ARNs or unique IDs as input.

## » Example Usage

```
data "aws_iam_user" "example" {
  user_name = "an_example_user_name"
}
```

## » Argument Reference

- `user_name` - (Required) The friendly IAM user name to match.

## » Attributes Reference

- `arn` - The Amazon Resource Name (ARN) assigned by AWS for this user.
- `path` - Path in which this user was created.
- `user_id` - The unique ID assigned by AWS for this user.

## » Data Source: `aws_inspector_rules_packages`

The AWS Inspector Rules Packages data source allows access to the list of AWS Inspector Rules Packages which can be used by AWS Inspector within the region configured in the provider.

## » Example Usage

```
# Declare the data source
data "aws_inspector_rules_packages" "rules" {}

# e.g. Use in aws_inspector_assessment_template
resource "aws_inspector_resource_group" "group" {
  tags {
    test = "test"
  }
}

resource "aws_inspector_assessment_target" "assessment" {
  name                = "test"
  resource_group_arn = "${aws_inspector_resource_group.group.arn}"
}

resource "aws_inspector_assessment_template" "assessment" {
```

```

name          = "Test"
target_arn    = "${aws_inspector_assessment_target.assessment.arn}"
duration      = "60"

rules_package_arns = "${data.aws_inspector_rules_packages.rules.arns}"
}

```

## » Attributes Reference

The following attributes are exported:

- **arns** - A list of the AWS Inspector Rules Packages arns available in the AWS region.

## » Data Source: `aws_instance`

Use this data source to get the ID of an Amazon EC2 Instance for use in other resources.

## » Example Usage

```

data "aws_instance" "foo" {
  instance_id = "i-instanceid"

  filter {
    name   = "image-id"
    values = ["ami-xxxxxxx"]
  }

  filter {
    name   = "tag:Name"
    values = ["instance-name-tag"]
  }
}

```

## » Argument Reference

- **instance\_id** - (Optional) Specify the exact Instance ID with which to populate the data source.
- **instance\_tags** - (Optional) A mapping of tags, each pair of which must exactly match a pair on the desired Instance.

- **filter** - (Optional) One or more name/value pairs to use as filters. There are several valid keys, for a full reference, check out describe-instances in the AWS CLI reference.
- **get\_password\_data** - (Optional) If true, wait for password data to become available and retrieve it. Useful for getting the administrator password for instances running Microsoft Windows. The password data is exported to the **password\_data** attribute. See GetPasswordData for more information.

**NOTE:** At least one of **filter**, **instance\_tags**, or **instance\_id** must be specified.

**NOTE:** If anything other than a single match is returned by the search, Terraform will fail. Ensure that your search is specific enough to return a single Instance ID only.

## » Attributes Reference

**id** is set to the ID of the found Instance. In addition, the following attributes are exported:

**NOTE:** Some values are not always set and may not be available for interpolation.

- **ami** - The ID of the AMI used to launch the instance.
- **associate\_public\_ip\_address** - Whether or not the Instance is associated with a public IP address or not (Boolean).
- **availability\_zone** - The availability zone of the Instance.
- **ebs\_block\_device** - The EBS block device mappings of the Instance.
  - **delete\_on\_termination** - If the EBS volume will be deleted on termination.
  - **device\_name** - The physical name of the device.
  - **encrypted** - If the EBS volume is encrypted.
  - **iops** - 0 If the EBS volume is not a provisioned IOPS image, otherwise the supported IOPS count.
  - **snapshot\_id** - The ID of the snapshot.
  - **volume\_size** - The size of the volume, in GiB.
  - **volume\_type** - The volume type.
- **ebs\_optimized** - Whether the Instance is EBS optimized or not (Boolean).
- **ephemeral\_block\_device** - The ephemeral block device mappings of the Instance.
  - **device\_name** - The physical name of the device.
  - **no\_device** - Whether the specified device included in the device mapping was suppressed or not (Boolean).
  - **virtual\_name** - The virtual device name.

- `iam_instance_profile` - The name of the instance profile associated with the Instance.
- `ipv6_addresses` - The IPv6 addresses associated to the Instance, if applicable. **NOTE:** Unlike the IPv4 address, this doesn't change if you attach an EIP to the instance.
- `instance_type` - The type of the Instance.
- `key_name` - The key name of the Instance.
- `monitoring` - Whether detailed monitoring is enabled or disabled for the Instance (Boolean).
- `network_interface_id` - The ID of the network interface that was created with the Instance.
- `password_data` - Base-64 encoded encrypted password data for the instance. Useful for getting the administrator password for instances running Microsoft Windows. This attribute is only exported if `get_password_data` is true. See `GetPasswordData` for more information.
- `placement_group` - The placement group of the Instance.
- `private_dns` - The private DNS name assigned to the Instance. Can only be used inside the Amazon EC2, and only available if you've enabled DNS hostnames for your VPC.
- `private_ip` - The private IP address assigned to the Instance.
- `public_dns` - The public DNS name assigned to the Instance. For EC2-VPC, this is only available if you've enabled DNS hostnames for your VPC.
- `public_ip` - The public IP address assigned to the Instance, if applicable. **NOTE:** If you are using an `aws_eip` with your instance, you should refer to the EIP's address directly and not use `public_ip`, as this field will change after the EIP is attached.
- `root_block_device` - The root block device mappings of the Instance
  - `delete_on_termination` - If the root block device will be deleted on termination.
  - `iops` - 0 If the volume is not a provisioned IOPS image, otherwise the supported IOPS count.
  - `volume_size` - The size of the volume, in GiB.
  - `volume_type` - The type of the volume.
- `security_groups` - The associated security groups.
- `source_dest_check` - Whether the network interface performs source/destination checking (Boolean).
- `subnet_id` - The VPC subnet ID.
- `user_data` - The User Data supplied to the Instance.
- `tags` - A mapping of tags assigned to the Instance.
- `tenancy` - The tenancy of the instance: `dedicated`, `default`, `host`.
- `vpc_security_group_ids` - The associated security groups in a non-default VPC.
- `credit_specification` - The credit specification of the Instance.



## » Data Source: `aws_instances`

Use this data source to get IDs or IPs of Amazon EC2 instances to be referenced elsewhere, e.g. to allow easier migration from another management solution or to make it easier for an operator to connect through bastion host(s).

**Note:** It's a best practice to expose instance details via outputs and remote state and **use `terraform_remote_state` data source instead** if you manage referenced instances via Terraform.

**Note:** It's strongly discouraged to use this data source for querying ephemeral instances (e.g. managed via autoscaling group), as the output may change at any time and you'd need to re-run `apply` every time an instance comes up or dies.

## » Example Usage

```
data "aws_instances" "test" {
  instance_tags {
    Role = "HardWorker"
  }
  filter {
    name   = "instance.group-id"
    values = ["sg-12345678"]
  }
}

resource "aws_eip" "test" {
  count = "${length(data.aws_instances.test.ids)}"
  instance = "${data.aws_instances.test.ids[count.index]}"
}
```

## » Argument Reference

- `instance_tags` - (Optional) A mapping of tags, each pair of which must exactly match a pair on desired instances.
- `filter` - (Optional) One or more name/value pairs to use as filters. There are several valid keys, for a full reference, check out `describe-instances` in the AWS CLI reference.

## » Attributes Reference

- `ids` - IDs of instances found through the filter

- `private_ips` - Private IP addresses of instances found through the filter
- `public_ips` - Public IP addresses of instances found through the filter

## » Data Source: `aws_internet_gateway`

`aws_internet_gateway` provides details about a specific Internet Gateway.

### » Example Usage

```
variable "vpc_id" {}

data "aws_internet_gateway" "default" {
  filter {
    name = "attachment.vpc-id"
    values = ["${var.vpc_id}"]
  }
}
```

### » Argument Reference

The arguments of this data source act as filters for querying the available Internet Gateway in the current region. The given filters must match exactly one Internet Gateway whose data will be exported as attributes.

- `internet_gateway_id` - (Optional) The id of the specific Internet Gateway to retrieve.
- `tags` - (Optional) A mapping of tags, each pair of which must exactly match a pair on the desired Internet Gateway.
- `filter` - (Optional) Custom filter block as described below.

More complex filters can be expressed using one or more `filter` sub-blocks, which take the following arguments:

- `name` - (Required) The name of the field to filter by, as defined by the underlying AWS API.
- `values` - (Required) Set of values that are accepted for the given field. An Internet Gateway will be selected if any one of the given values matches.

### » Attributes Reference

All of the argument attributes except `filter` block are also exported as result attributes. This data source will complete the data by populating any fields

that are not included in the configuration with the data for the selected Internet Gateway.

**attachments** are also exported with the following attributes, when there are relevant: Each attachment supports the following:

- **state** - The current state of the attachment between the gateway and the VPC. Present only if a VPC is attached
- **vpc\_id** - The ID of an attached VPC.

## » Data Source: `aws__iot__endpoint`

Returns a unique endpoint specific to the AWS account making the call.

### » Example Usage

```
data "aws_iot_endpoint" "example" {}

resource "kubernetes_pod" "agent" {
  metadata {
    name = "my-device"
  }
  spec {
    container {
      image = "gcr.io/my-project/image-name"
      name  = "image-name"

      env = [
        {
          name = "IOT_ENDPOINT"
          value = "${data.aws_iot_endpoint.example.endpoint_address}"
        },
      ]
    }
  }
}
```

### » Argument Reference

N/A

## » Attributes Reference

- `endpoint_address` - The endpoint. The format of the endpoint is as follows: `IDENTIFIER.iot.REGION.amazonaws.com`.

## » Data Source: `aws_ip_ranges`

Use this data source to get the IP ranges of various AWS products and services.

## » Example Usage

```
data "aws_ip_ranges" "european_ec2" {
  regions = ["eu-west-1", "eu-central-1"]
  services = ["ec2"]
}

resource "aws_security_group" "from_europe" {
  name = "from_europe"

  ingress {
    from_port   = "443"
    to_port     = "443"
    protocol    = "tcp"
    cidr_blocks = ["${data.aws_ip_ranges.european_ec2.cidr_blocks}"]
  }

  tags {
    CreateDate = "${data.aws_ip_ranges.european_ec2.create_date}"
    SyncToken  = "${data.aws_ip_ranges.european_ec2.sync_token}"
  }
}
```

## » Argument Reference

- `regions` - (Optional) Filter IP ranges by regions (or include all regions, if omitted). Valid items are `global` (for `cloudfront`) as well as all AWS regions (e.g. `eu-central-1`)
- `services` - (Required) Filter IP ranges by services. Valid items are `amazon` (for `amazon.com`), `cloudfront`, `codebuild`, `ec2`, `route53`, `route53_healthchecks` and `S3`.

**NOTE:** If the specified combination of regions and services does not yield any CIDR blocks, Terraform will fail.

## » Attributes Reference

- `cidr_blocks` - The lexically ordered list of CIDR blocks.
- `create_date` - The publication time of the IP ranges (e.g. 2016-08-03-23-46-05).
- `sync_token` - The publication time of the IP ranges, in Unix epoch time format (e.g. 1470267965).

## » Data Source: `aws_kinesis_stream`

Use this data source to get information about a Kinesis Stream for use in other resources.

For more details, see the Amazon Kinesis Documentation.

## » Example Usage

```
data "aws_kinesis_stream" "stream" {  
  name = "stream-name"  
}
```

## » Argument Reference

- `name` - (Required) The name of the Kinesis Stream.

## » Attributes Reference

`id` is set to the Amazon Resource Name (ARN) of the Kinesis Stream. In addition, the following attributes are exported:

- `arn` - The Amazon Resource Name (ARN) of the Kinesis Stream (same as `id`).
- `name` - The name of the Kinesis Stream.
- `creation_timestamp` - The approximate UNIX timestamp that the stream was created.
- `status` - The current status of the stream. The stream status is one of CREATING, DELETING, ACTIVE, or UPDATING.
- `retention_period` - Length of time (in hours) data records are accessible after they are added to the stream.

- **open\_shards** - The list of shard ids in the OPEN state. See Shard State for more.
- **closed\_shards** - The list of shard ids in the CLOSED state. See Shard State for more.
- **shard\_level\_metrics** - A list of shard-level CloudWatch metrics which are enabled for the stream. See Monitoring with CloudWatch for more.
- **tags** - A mapping of tags to assigned to the stream.

## » Data Source: **aws\_kms\_alias**

Use this data source to get the ARN of a KMS key alias. By using this data source, you can reference key alias without having to hard code the ARN as input.

### » Example Usage

```
data "aws_kms_alias" "s3" {
  name = "alias/aws/s3"
}
```

### » Argument Reference

- **name** - (Required) The display name of the alias. The name must start with the word "alias" followed by a forward slash (alias/)

### » Attributes Reference

- **arn** - The Amazon Resource Name(ARN) of the key alias.
- **target\_key\_id** - Key identifier pointed to by the alias.
- **target\_key\_arn** - ARN pointed to by the alias.

## » **aws\_kms\_key**

Use this data source to get detailed information about the specified KMS Key with flexible key id input. This can be useful to reference key alias without having to hard code the ARN as input.

## » Example Usage

```
data "aws_kms_key" "foo" {  
  key_id = "alias/my-key"  
}
```

```
data "aws_kms_key" "foo" {  
  key_id = "1234abcd-12ab-34cd-56ef-1234567890ab"  
}
```

```
data "aws_kms_key" "foo" {  
  key_id = "arn:aws:kms:us-east-1:111122223333:alias/my-key"  
}
```

```
data "aws_kms_key" "foo" {  
  key_id = "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"  
}
```

## » Argument Reference

- **key\_id** - (Required) Key identifier which can be one of the following format:
  - Key ID. E.g: 1234abcd-12ab-34cd-56ef-1234567890ab
  - Key ARN. E.g.: arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
  - Alias name. E.g.: alias/my-key
  - Alias ARN. E.g.: arn:aws:kms:us-east-1:111122223333:alias/my-key
- **grant\_tokens** - (Optional) List of grant tokens

## » Attributes Reference

- **id**: The globally unique identifier for the key
- **arn**: The Amazon Resource Name (ARN) of the key
- **aws\_account\_id**: The twelve-digit account ID of the AWS account that owns the key
- **creation\_date**: The date and time when the key was created
- **deletion\_date**: The date and time after which AWS KMS deletes the key. This value is present only when **key\_state** is **PendingDeletion**, otherwise this value is 0
- **description**: The description of the key.
- **enabled**: Specifies whether the key is enabled. When **key\_state** is **Enabled** this value is true, otherwise it is false
- **expiration\_model**: Specifies whether the Key's key material expires. This value is present only when **origin** is **EXTERNAL**, otherwise this value is empty

- **key\_manager:** The key's manager
- **key\_state:** The state of the key
- **key\_usage:** Currently the only allowed value is `ENCRYPT_DECRYPT`
- **origin:** When this value is `AWS_KMS`, AWS KMS created the key material. When this value is `EXTERNAL`, the key material was imported from your existing key management infrastructure or the CMK lacks key material
- **valid\_to:** The time at which the imported key material expires. This value is present only when **origin** is `EXTERNAL` and whose **expiration\_model** is `KEY_MATERIAL_EXPIRES`, otherwise this value is 0

## » Data Source: `aws_kms_ciphertext`

The KMS ciphertext data source allows you to encrypt plaintext into ciphertext by using an AWS KMS customer master key.

**Note:** All arguments including the plaintext be stored in the raw state as plain-text. Read more about sensitive data in state.

### » Example Usage

```
resource "aws_kms_key" "oauth_config" {
  description = "oauth config"
  is_enabled = true
}

data "aws_kms_ciphertext" "oauth" {
  key_id = "${aws_kms_key.oauth_config.key_id}"
  plaintext = <<EOF
{
  "client_id": "e587dbae22222f55da22",
  "client_secret": "8289575d00000ace55e1815ec13673955721b8a5"
}
EOF
}
```

### » Argument Reference

The following arguments are supported:

- **plaintext** - (Required) Data to be encrypted. Note that this may show up in logs, and it will be stored in the state file.
- **key\_id** - (Required) Globally unique key ID for the customer master key.



- **context** - (Optional) An optional mapping that makes up the encryption context.

## » Attributes Reference

All of the argument attributes are also exported as result attributes.

- **ciphertext\_blob** - Base64 encoded ciphertext

## » Data Source: `aws_kms_secret`

The KMS secret data source allows you to use data encrypted with the AWS KMS service within your resource definitions.

**NOTE:** Using this data provider will allow you to conceal secret data within your resource definitions but does not take care of protecting that data in the logging output, plan output or state output.

Please take care to secure your secret data outside of resource definitions.

## » Example Usage

First, let's encrypt a password with KMS using the AWS CLI tools. This requires you to have your AWS CLI setup correctly, and you would replace the key-id with your own. If you have a newline character at the end of your file, secrets will be decrypted with this newline character intact. For most use-cases this is undesirable and leads to incorrect passwords or invalid values, as well as possible changes in the plan. Alternatively you can use `--plaintext 'password'` instead of reading from a file.

```
$ echo -n 'master-password' > plaintext-password
$ aws kms encrypt \
> --key-id ab123456-c012-4567-890a-deadbeef123 \
> --plaintext fileb://plaintext-password \
> --encryption-context foo=bar \
> --output text --query CiphertextBlob
AQECAHgaPa0J8WadplGCqQVAr4HNvDaFSQ+NaiwIBhmm6qDSFwAAAGIwYAYJKoZIhvcNAQcGoFMwUQIBADBMBGkqhki
```

Now, take that output and add it to your resource definitions.

```
data "aws_kms_secret" "db" {
  secret {
    name      = "master_password"
    payload   = "AQECAHgaPa0J8WadplGCqQVAr4HNvDaFSQ+NaiwIBhmm6qDSFwAAAGIwYAYJKoZIhvcNAQcGoFMwUQIBADBMBGkqhki"
```

```

    context {
        foo = "bar"
    }
}

resource "aws_rds_cluster" "rds" {
    master_username = "root"
    master_password = "${data.aws_kms_secret.db.master_password}"

    # ...
}

```

And your RDS cluster would have the root password set to "master-password"

## » Argument Reference

The following arguments are supported:

- **secret** - (Required) One or more encrypted payload definitions from the KMS service. See the Secret Definitions below.

## » Secret Definitions

Each secret definition supports the following arguments:

- **name** - (Required) The name to export this secret under in the attributes.
- **payload** - (Required) Base64 encoded payload, as returned from a KMS encrypt operation.
- **context** - (Optional) An optional mapping that makes up the Encryption Context for the secret.
- **grant\_tokens** (Optional) An optional list of Grant Tokens for the secret.

For more information on **context** and **grant\_tokens** see the KMS Concepts

## » Attributes Reference

Each **secret** defined is exported under its **name** as a top-level attribute.

## » Data Source: `aws_nat_gateway`

Provides details about a specific Nat Gateway.

## » Example Usage

```
variable "subnet_id" {}

data "aws_nat_gateway" "default" {
  subnet_id = "${aws_subnet.public.id}"
}
```

Usage with tags:

```
data "aws_nat_gateway" "default" {
  subnet_id = "${aws_subnet.public.id}"

  tags {
    Name = "gw NAT"
  }
}
```

## » Argument Reference

The arguments of this data source act as filters for querying the available Nat Gateways in the current region. The given filters must match exactly one Nat Gateway whose data will be exported as attributes.

- **id** - (Optional) The id of the specific Nat Gateway to retrieve.
- **subnet\_id** - (Optional) The id of subnet that the Nat Gateway resides in.
- **vpc\_id** - (Optional) The id of the VPC that the Nat Gateway resides in.
- **state** - (Optional) The state of the NAT gateway (pending | failed | available | deleting | deleted ).
- **filter** - (Optional) Custom filter block as described below. More complex filters can be expressed using one or more **filter** sub-blocks, which take the following arguments:
- **name** - (Required) The name of the field to filter by, as defined by the underlying AWS API.
- **values** - (Required) Set of values that are accepted for the given field. An Nat Gateway will be selected if any one of the given values matches.

## » Attributes Reference

All of the argument attributes except **filter** block are also exported as result attributes. This data source will complete the data by populating any fields that are not included in the configuration with the data for the selected Nat Gateway.

**addresses** are also exported with the following attributes, when they are relevant: Each attachment supports the following:

- `allocation_id` - The Id of the EIP allocated to the selected Nat Gateway.
- `network_interface_id` - The Id of the ENI allocated to the selected Nat Gateway.
- `private_ip` - The private Ip address of the selected Nat Gateway.
- `public_ip` - The public Ip (EIP) address of the selected Nat Gateway.

## » `aws__network__interface`

Use this data source to get information about a Network Interface.

### » Example Usage

```
data "aws_network_interface" "bar" {
  id = "eni-01234567"
}
```

### » Argument Reference

The following arguments are supported:

- `id` - (Optional) The identifier for the network interface.
- `filter` - (Optional) One or more name/value pairs to filter off of. There are several valid keys, for a full reference, check out `describe-network-interfaces` in the AWS CLI reference.

### » Attributes Reference

See the Network Interface for details on the returned attributes.

Additionally, the following attributes are exported:

- `association` - The association information for an Elastic IP address (IPv4) associated with the network interface. See supported fields below.
- `availability_zone` - The Availability Zone.
- `interface_type` - The type of interface.
- `ipv6_addresses` - List of IPv6 addresses to assign to the ENI.
- `mac_address` - The MAC address.
- `owner_id` - The AWS account ID of the owner of the network interface.
- `requester_id` - The ID of the entity that launched the instance on your behalf.

#### » **association**

- `allocation_id` - The allocation ID.
- `association_id` - The association ID.
- `ip_owner_id` - The ID of the Elastic IP address owner.
- `public_dns_name` - The public DNS name.
- `public_ip` - The address of the Elastic IP address bound to the network interface.

#### » **Import**

Elastic Network Interfaces can be imported using the `id`, e.g.

```
$ terraform import aws_network_interface.test eni-12345
```

#### » **aws\_lambda\_function**

Provides information about a Lambda Function.

#### » **Example Usage**

```
variable "function_name" {  
  type = "string"  
}  
  
data "aws_lambda_function" "existing" {  
  function_name = "${var.function_name}"  
}
```

#### » **Argument Reference**

The following arguments are supported:

- `function_name` - (Required) Name of the lambda function.
- `qualifier` - (Optional) Qualifier of the lambda function. Defaults to `$LATEST`.

#### » **Attributes Reference**

The following attributes are exported:

- **arn** - The Amazon Resource Name (ARN) identifying your Lambda Function.
- **dead\_letter\_config** - Configure the function's *dead letter queue*.
- **description** - Description of what your Lambda Function does.
- **environment** - The Lambda environment's configuration settings.
- **handler** - The function entrypoint in your code.
- **invoke\_arn** - The ARN to be used for invoking Lambda Function from API Gateway.
- **kms\_key\_arn** - The ARN for the KMS encryption key.
- **last\_modified** - The date this resource was last modified.
- **memory\_size** - Amount of memory in MB your Lambda Function can use at runtime.
- **qualified\_arn** - The Amazon Resource Name (ARN) identifying your Lambda Function Version
- **reserved\_concurrent\_executions** - The amount of reserved concurrent executions for this lambda function.
- **role** - IAM role attached to the Lambda Function.
- **runtime** - The runtime environment for the Lambda function..
- **source\_code\_hash** - Base64-encoded representation of raw SHA-256 sum of the zip file.
- **source\_code\_size** - The size in bytes of the function .zip file.
- **timeout** - The function execution time at which Lambda should terminate the function.
- **tracing\_config** - Tracing settings of the function.
- **version** - The version of the Lambda function.
- **vpc\_config** - VPC configuration associated with your Lambda function.

## » Data Source: `aws_lb`

**Note:** `aws_alb` is known as `aws_lb`. The functionality is identical.

Provides information about a Load Balancer.

This data source can prove useful when a module accepts an LB as an input variable and needs to, for example, determine the security groups associated with it, etc.

## » Example Usage

```
variable "lb_arn" {
  type    = "string"
  default = ""
}
```

```
variable "lb_name" {
  type    = "string"
  default = ""
}

data "aws_lb" "test" {
  arn  = "${var.lb_arn}"
  name = "${var.lb_name}"
}
```

## » Argument Reference

The following arguments are supported:

- **arn** - (Optional) The full ARN of the load balancer.
- **name** - (Optional) The unique name of the load balancer.

**NOTE:** When both **arn** and **name** are specified, **arn** takes precedence.

## » Attributes Reference

See the LB Resource for details on the returned attributes - they are identical.

## » Data Source: `aws_lb_listener`

**Note:** `aws_alb_listener` is known as `aws_lb_listener`. The functionality is identical.

Provides information about a Load Balancer Listener.

This data source can prove useful when a module accepts an LB Listener as an input variable and needs to know the LB it is attached to, or other information specific to the listener in question.

## » Example Usage

```
# get listener from listener arn

variable "listener_arn" {
  type = "string"
}

data "aws_lb_listener" "listener" {
```

```

    arn = "${var.listener_arn}"
  }

# get listener from load_balancer_arn and port

data "aws_lb" "selected" {
  name = "default-public"
}

data "aws_lb_listener" "selected443" {
  load_balancer_arn = "${data.aws_lb.selected.arn}"
  port = 443
}

```

## » Argument Reference

The following arguments are supported:

- **arn** - (Optional) The arn of the listener. Required if **load\_balancer\_arn** and **port** is not set.
- **load\_balancer\_arn** - (Optional) The arn of the load balancer. Required if **arn** is not set.
- **port** - (Optional) The port of the listener. Required if **arn** is not set.

## » Attributes Reference

See the LB Listener Resource for details on the returned attributes - they are identical.

## » Data Source: `aws_lb_target_group`

**Note:** `aws_alb_target_group` is known as `aws_lb_target_group`. The functionality is identical.

Provides information about a Load Balancer Target Group.

This data source can prove useful when a module accepts an LB Target Group as an input variable and needs to know its attributes. It can also be used to get the ARN of an LB Target Group for use in other resources, given LB Target Group name.



## » Example Usage

```
variable "lb_tg_arn" {
  type    = "string"
  default = ""
}

variable "lb_tg_name" {
  type    = "string"
  default = ""
}

data "aws_lb_target_group" "test" {
  arn    = "${var.lb_tg_arn}"
  name   = "${var.lb_tg_name}"
}
```

## » Argument Reference

The following arguments are supported:

- **arn** - (Optional) The full ARN of the target group.
- **name** - (Optional) The unique name of the target group.

**NOTE:** When both **arn** and **name** are specified, **arn** takes precedence.

## » Attributes Reference

See the LB Target Group Resource for details on the returned attributes - they are identical.

## » Data Source: `aws__mq__broker`

Provides information about a MQ Broker.

## » Example Usage

```
variable "broker_id" {
  type    = "string"
  default = ""
}
```

```

variable "broker_name" {
  type    = "string"
  default = ""
}

data "aws_mq_broker" "by_id" {
  broker_id = "${var.broker_id}"
}

data "aws_mq_broker" "by_name" {
  broker_name = "${var.broker_name}"
}

```

## » Argument Reference

The following arguments are supported:

- `broker_id` - (Optional) The unique id of the mq broker.
- `broker_name` - (Optional) The unique name of the mq broker.

## » Attributes Reference

See the `aws_mq_broker` resource for details on the returned attributes. They are identical except for user password, which is not returned when describing broker.

## » Data Source: `aws_partition`

Use this data source to lookup current AWS partition in which Terraform is working

## » Example Usage

```

data "aws_partition" "current" {}

data "aws_iam_policy_document" "s3_policy" {
  statement {
    sid = "1"

    actions = [
      "s3:ListBucket",
    ]
  }
}

```

```

    resources = [
        "arn:${data.aws_partition.current.partition}:s3::my-bucket",
    ]
}

```

## » Argument Reference

There are no arguments available for this data source.

## » Attributes Reference

`partition` is set to the identifier of the current partition.

## » Data Source: `aws_prefix_list`

`aws_prefix_list` provides details about a specific prefix list (PL) in the current region.

This can be used both to validate a prefix list given in a variable and to obtain the CIDR blocks (IP address ranges) for the associated AWS service. The latter may be useful e.g. for adding network ACL rules.

## » Example Usage

```

resource "aws_vpc_endpoint" "private_s3" {
    vpc_id      = "${aws_vpc.foo.id}"
    service_name = "com.amazonaws.us-west-2.s3"
}

data "aws_prefix_list" "private_s3" {
    prefix_list_id = "${aws_vpc_endpoint.private_s3.prefix_list_id}"
}

resource "aws_network_acl" "bar" {
    vpc_id = "${aws_vpc.foo.id}"
}

resource "aws_network_acl_rule" "private_s3" {
    network_acl_id = "${aws_network_acl.bar.id}"
    rule_number    = 200
}

```

```

    egress          = false
    protocol        = "tcp"
    rule_action     = "allow"
    cidr_block      = "${data.aws_prefix_list.private_s3.cidr_blocks[0]}"
    from_port       = 443
    to_port         = 443
  }

```

## » Argument Reference

The arguments of this data source act as filters for querying the available prefix lists. The given filters must match exactly one prefix list whose data will be exported as attributes.

- `prefix_list_id` - (Optional) The ID of the prefix list to select.
- `name` - (Optional) The name of the prefix list to select.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the selected prefix list.
- `name` - The name of the selected prefix list.
- `cidr_blocks` - The list of CIDR blocks for the AWS service associated with the prefix list.

## » Data Source: `aws_rds_cluster`

Provides information about a RDS cluster.

## » Example Usage

```

data "aws_rds_cluster" "clusterName" {
  cluster_identifier = "clusterName"
}

```

## » Argument Reference

The following arguments are supported:

- `cluster_identifier` - (Required) The cluster identifier of the RDS cluster.

## » Attributes Reference

See the RDS Cluster Resource for details on the returned attributes - they are identical.

## » Data Source: `aws_redshift_cluster`

Provides details about a specific redshift cluster.

## » Example Usage

```
data "aws_redshift_cluster" "test_cluster" {
  cluster_identifier = "test-cluster"
}

resource "aws_kinesis_firehose_delivery_stream" "test_stream" {
  name          = "terraform-kinesis-firehose-test-stream"
  destination   = "redshift"

  s3_configuration {
    role_arn          = "${aws_iam_role.firehose_role.arn}"
    bucket_arn        = "${aws_s3_bucket.bucket.arn}"
    buffer_size       = 10
    buffer_interval   = 400
    compression_format = "GZIP"
  }

  redshift_configuration {
    role_arn          = "${aws_iam_role.firehose_role.arn}"
    cluster_jdbcurl    = "jdbc:redshift://${data.aws_redshift_cluster.test_cluster.endpoint}"
    username          = "testuser"
    password          = "T3stPass"
    data_table_name    = "test-table"
    copy_options       = "delimiter '|' # the default delimiter"
    data_table_columns = "test-col"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `cluster_identifier` - (Required) The cluster identifier

## » Attribute Reference

The following attributes are exported:

- `allow_version_upgrade` - Whether major version upgrades can be applied during maintenance period
- `automated_snapshot_retention_period` - The backup retention period
- `availability_zone` - The availability zone of the cluster
- `bucket_name` - The name of the S3 bucket where the log files are to be stored
- `cluster_identifier` - The cluster identifier
- `cluster_parameter_group_name` - The name of the parameter group to be associated with this cluster
- `cluster_public_key` - The public key for the cluster
- `cluster_revision_number` - The cluster revision number
- `cluster_security_groups` - The security groups associated with the cluster
- `cluster_subnet_group_name` - The name of a cluster subnet group to be associated with this cluster
- `cluster_type` - The cluster type
- `database_name` - The name of the default database in the cluster
- `elastic_ip` - The Elastic IP of the cluster
- `enable_logging` - Whether cluster logging is enabled
- `encrypted` - Whether the cluster data is encrypted
- `endpoint` - The cluster endpoint
- `enhanced_vpc_routing` - Whether enhanced VPC routing is enabled
- `iam_roles` - The IAM roles associated to the cluster
- `kms_key_id` - The KMS encryption key associated to the cluster
- `master_username` - Username for the master DB user
- `node_type` - The cluster node type
- `number_of_nodes` - The number of nodes in the cluster
- `port` - The port the cluster responds on
- `preferred_maintenance_window` - The maintenance window
- `publicly_accessible` - Whether the cluster is publicly accessible
- `s3_key_prefix` - The folder inside the S3 bucket where the log files are stored
- `tags` - The tags associated to the cluster
- `vpc_id` - The VPC Id associated with the cluster
- `vpc_security_group_ids` - The VPC security group Ids associated with the cluster

## » Data Source: `aws_redshift_service_account`

Use this data source to get the Account ID of the AWS Redshift Service Account in a given region for the purpose of allowing Redshift to store audit data in S3.

## » Example Usage

```
data "aws_redshift_service_account" "main" {}

resource "aws_s3_bucket" "bucket" {
  bucket          = "tf-redshift-logging-test-bucket"
  force_destroy   = true

  policy = <<EOF
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "Put bucket policy needed for audit logging",
      "Effect": "Allow",
      "Principal": {
        "AWS": "${data.aws_redshift_service_account.main.arn}"
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::tf-redshift-logging-test-bucket/*"
    },
    {
      "Sid": "Get bucket policy needed for audit logging ",
      "Effect": "Allow",
      "Principal": {
        "AWS": "${data.aws_redshift_service_account.main.arn}"
      },
      "Action": "s3:GetBucketAcl",
      "Resource": "arn:aws:s3:::tf-redshift-logging-test-bucket"
    }
  ]
}
EOF
}
```

## » Argument Reference

- `region` - (Optional) Name of the region whose AWS Redshift account ID is desired. Defaults to the region from the AWS provider configuration.

## » Attributes Reference

- `id` - The ID of the AWS Redshift service account in the selected region.
- `arn` - The ARN of the AWS Redshift service account in the selected region.

## » Data Source: `aws__region`

`aws_region` provides details about a specific AWS region.

As well as validating a given region name this resource can be used to discover the name of the region configured within the provider. The latter can be useful in a child module which is inheriting an AWS provider configuration from its parent module.

## » Example Usage

The following example shows how the resource might be used to obtain the name of the AWS region configured on the provider.

```
data "aws_region" "current" {}
```

## » Argument Reference

The arguments of this data source act as filters for querying the available regions. The given filters must match exactly one region whose data will be exported as attributes.

- `name` - (Optional) The full name of the region to select.
- `endpoint` - (Optional) The EC2 endpoint of the region to select.

## » Attributes Reference

The following attributes are exported:

- `name` - The name of the selected region.
- `current` - `true` if the selected region is the one configured on the provider, or `false` otherwise.
- `endpoint` - The EC2 endpoint for the selected region.



## » Data Source: `aws_route53_zone`

`aws_route53_zone` provides details about a specific Route 53 Hosted Zone.

This data source allows to find a Hosted Zone ID given Hosted Zone name and certain search criteria.

### » Example Usage

The following example shows how to get a Hosted Zone from its name and from this data how to create a Record Set.

```
data "aws_route53_zone" "selected" {
  name          = "test.com."
  private_zone = true
}

resource "aws_route53_record" "www" {
  zone_id = "${data.aws_route53_zone.selected.zone_id}"
  name     = "www.${data.aws_route53_zone.selected.name}"
  type     = "A"
  ttl      = "300"
  records  = ["10.0.0.1"]
}
```

### » Argument Reference

The arguments of this data source act as filters for querying the available Hosted Zone. You have to use `zone_id` or `name`, not both of them. The given filter must match exactly one Hosted Zone. If you use `name` field for private Hosted Zone, you need to add `private_zone` field to `true`

- `zone_id` - (Optional) The Hosted Zone id of the desired Hosted Zone.
- `name` - (Optional) The Hosted Zone name of the desired Hosted Zone.
- `private_zone` - (Optional) Used with `name` field to get a private Hosted Zone.
- `vpc_id` - (Optional) Used with `name` field to get a private Hosted Zone associated with the `vpc_id` (in this case, `private_zone` is not mandatory).
- `tags` - (Optional) Used with `name` field. A mapping of tags, each pair of which must exactly match a pair on the desired Hosted Zone.

## » Attributes Reference

All of the argument attributes are also exported as result attributes. This data source will complete the data by populating any fields that are not included in the configuration with the data for the selected Hosted Zone.

The following attribute is additionally exported:

- **caller\_reference** - Caller Reference of the Hosted Zone.
- **comment** - The comment field of the Hosted Zone.
- **name\_servers** - The list of DNS name servers for the Hosted Zone.
- **resource\_record\_set\_count** - the number of Record Set in the Hosted Zone

## » Data Source: `aws_route_table`

`aws_route_table` provides details about a specific Route Table.

This resource can prove useful when a module accepts a Subnet id as an input variable and needs to, for example, add a route in the Route Table.

## » Example Usage

The following example shows how one might accept a Route Table id as a variable and use this data source to obtain the data necessary to create a route.

```
variable "subnet_id" {}

data "aws_route_table" "selected" {
  subnet_id = "${var.subnet_id}"
}

resource "aws_route" "route" {
  route_table_id      = "${data.aws_route_table.selected.id}"
  destination_cidr_block = "10.0.1.0/22"
  vpc_peering_connection_id = "pcx-45ff3dc1"
}
```

## » Argument Reference

The arguments of this data source act as filters for querying the available Route Table in the current region. The given filters must match exactly one Route Table whose data will be exported as attributes.

- **filter** - (Optional) Custom filter block as described below.

- **route\_table\_id** - (Optional) The id of the specific Route Table to retrieve.
- **tags** - (Optional) A mapping of tags, each pair of which must exactly match a pair on the desired Route Table.
- **vpc\_id** - (Optional) The id of the VPC that the desired Route Table belongs to.
- **subnet\_id** - (Optional) The id of a Subnet which is connected to the Route Table (not be exported if not given in parameter).

More complex filters can be expressed using one or more **filter** sub-blocks, which take the following arguments:

- **name** - (Required) The name of the field to filter by, as defined by the underlying AWS API.
- **values** - (Required) Set of values that are accepted for the given field. A Route Table will be selected if any one of the given values matches.

## » Attributes Reference

All of the argument attributes except **filter** and **subnet\_id** blocks are also exported as result attributes. This data source will complete the data by populating any fields that are not included in the configuration with the data for the selected Route Table.

**routes** are also exported with the following attributes, when there are relevant: Each route supports the following:

- **cidr\_block** - The CIDR block of the route.
- **ipv6\_cidr\_block** - The IPv6 CIDR block of the route.
- **egress\_only\_gateway\_id** - The ID of the Egress Only Internet Gateway.
- **gateway\_id** - The Internet Gateway ID.
- **nat\_gateway\_id** - The NAT Gateway ID.
- **instance\_id** - The EC2 instance ID.
- **vpc\_peering\_connection\_id** - The VPC Peering ID.
- **network\_interface\_id** - The ID of the elastic network interface (eni) to use.

**associations** are also exported with the following attributes:

- **route\_table\_association\_id** - The Association ID .
- **route\_table\_id** - The Route Table ID.
- **subnet\_id** - The Subnet ID.
- **main** - If the Association due to the Main Route Table.

## » Data Source: aws\_s3\_bucket

Provides details about a specific S3 bucket.

This resource may prove useful when setting up a Route53 record, or an origin for a CloudFront Distribution.

## » Example Usage

### » Route53 Record

```
data "aws_s3_bucket" "selected" {
  bucket = "bucket.test.com"
}

data "aws_route53_zone" "test_zone" {
  name = "test.com."
}

resource "aws_route53_record" "example" {
  zone_id = "${data.aws_route53_zone.test_zone.id}"
  name    = "bucket"
  type    = "A"

  alias {
    name      = "${data.aws_s3_bucket.selected.website_domain}"
    zone_id = "${data.aws_s3_bucket.selected.hosted_zone_id}"
  }
}
```

### » CloudFront Origin

```
data "aws_s3_bucket" "selected" {
  bucket = "a-test-bucket"
}

resource "aws_cloudfront_distribution" "test" {
  origin {
    domain_name = "${data.aws_s3_bucket.selected.bucket_domain_name}"
    origin_id   = "s3-selected-bucket"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **bucket** - (Required) The name of the bucket

## » Attribute Reference

The following attributes are exported:

- **id** - The name of the bucket.
- **arn** - The ARN of the bucket. Will be of format `arn:aws:s3:::bucketname`.
- **bucket\_domain\_name** - The bucket domain name. Will be of format `bucketname.s3.amazonaws.com`.
- **hosted\_zone\_id** - The Route 53 Hosted Zone ID for this bucket's region.
- **region** - The AWS region this bucket resides in.
- **website\_endpoint** - The website endpoint, if the bucket is configured with a website. If not, this will be an empty string.
- **website\_domain** - The domain of the website endpoint, if the bucket is configured with a website. If not, this will be an empty string. This is used to create Route 53 alias records.

## » Data Source: `aws_s3_bucket_object`

The S3 object data source allows access to the metadata and *optionally* (see below) content of an object stored inside S3 bucket.

**Note:** The content of an object (`body` field) is available only for objects which have a human-readable **Content-Type** (`text/*` and `application/json`). This is to prevent printing unsafe characters and potentially downloading large amount of data which would be thrown away in favour of metadata.

## » Example Usage

The following example retrieves a text object (which must have a **Content-Type** value starting with `text/`) and uses it as the `user_data` for an EC2 instance:

```
data "aws_s3_bucket_object" "bootstrap_script" {
  bucket = "ourcorp-deploy-config"
  key    = "ec2-bootstrap-script.sh"
}
```

```
resource "aws_instance" "example" {
  instance_type = "t2.micro"
```

```

ami          = "ami-2757f631"
user_data    = "${data.aws_s3_bucket_object.bootstrap_script.body}"
}

```

The following, more-complex example retrieves only the metadata for a zip file stored in S3, which is then used to pass the most recent `version_id` to AWS Lambda for use as a function implementation. More information about Lambda functions is available in the documentation for `aws_lambda_function`.

```

data "aws_s3_bucket_object" "lambda" {
  bucket = "ourcorp-lambda-functions"
  key     = "hello-world.zip"
}

resource "aws_lambda_function" "test_lambda" {
  s3_bucket      = "${data.aws_s3_bucket_object.lambda.bucket}"
  s3_key          = "${data.aws_s3_bucket_object.lambda.key}"
  s3_object_version = "${data.aws_s3_bucket_object.lambda.version_id}"
  function_name   = "lambda_function_name"
  role            = "${aws_iam_role.iam_for_lambda.arn}" # (not shown)
  handler         = "exports.test"
}

```

## » Argument Reference

The following arguments are supported:

- `bucket` - (Required) The name of the bucket to read the object from
- `key` - (Required) The full path to the object inside the bucket
- `version_id` - (Optional) Specific version ID of the object returned (defaults to latest version)

## » Attributes Reference

The following attributes are exported:

- `body` - Object data (see **limitations above** to understand cases in which this field is actually available)
- `cache_control` - Specifies caching behavior along the request/reply chain.
- `content_disposition` - Specifies presentational information for the object.
- `content_encoding` - Specifies what content encodings have been applied to the object and thus what decoding mechanisms must be applied to obtain the media-type referenced by the Content-Type header field.
- `content_language` - The language the content is in.
- `content_length` - Size of the body in bytes.

- **content\_type** - A standard MIME type describing the format of the object data.
- **etag** - ETag generated for the object (an MD5 sum of the object content in case it's not encrypted)
- **expiration** - If the object expiration is configured (see object lifecycle management), the field includes this header. It includes the expiry-date and rule-id key value pairs providing object expiration information. The value of the rule-id is URL encoded.
- **expires** - The date and time at which the object is no longer cacheable.
- **last\_modified** - Last modified date of the object in RFC1123 format (e.g. Mon, 02 Jan 2006 15:04:05 MST)
- **metadata** - A map of metadata stored with the object in S3
- **server\_side\_encryption** - If the object is stored using server-side encryption (KMS or Amazon S3-managed encryption key), this field includes the chosen encryption and algorithm used.
- **sse\_kms\_key\_id** - If present, specifies the ID of the Key Management Service (KMS) master encryption key that was used for the object.
- **storage\_class** - Storage class information of the object. Available for all objects except for **Standard** storage class objects.
- **version\_id** - The latest version ID of the object returned.
- **website\_redirect\_location** - If the bucket is configured as a website, redirects requests for this object to another object in the same bucket or to an external URL. Amazon S3 stores the value of this header in the object metadata.
- **tags** - A mapping of tags assigned to the object.

## » Data Source: `aws_secretsmanager_secret`

Retrieve metadata information about a Secrets Manager secret. To retrieve a secret value, see the `aws_secretsmanager_secret_version` data source.

### » Example Usage

#### » ARN

```
data "aws_secretsmanager_secret" "by-arn" {
  arn = "arn:aws:secretsmanager:us-east-1:123456789012:secret:example-123456"
}
```

#### » Name

```
data "aws_secretsmanager_secret" "by-name" {
```

```

    name = "example"
}

```

## » Argument Reference

- **arn** - (Optional) The Amazon Resource Name (ARN) of the secret to retrieve.
- **name** - (Optional) The name of the secret to retrieve.

## » Attributes Reference

- **arn** - The Amazon Resource Name (ARN) of the secret.
- **description** - A description of the secret.
- **kms\_key\_id** - The Key Management Service (KMS) Customer Master Key (CMK) associated with the secret.
- **id** - The Amazon Resource Name (ARN) of the secret.
- **rotation\_enabled** - Whether rotation is enabled or not.
- **rotation\_lambda\_arn** - Rotation Lambda function Amazon Resource Name (ARN) if rotation is enabled.
- **rotation\_rules** - Rotation rules if rotation is enabled.
- **tags** - Tags of the secret.

## » Data Source: `aws_secretsmanager_secret_version`

Retrieve information about a Secrets Manager secret version including its secret value. To retrieve secret metadata, see the `aws_secretsmanager_secret` data source.

## » Example Usage

### » Retrieve Current Secret Version

By default, this data sources retrieves information based on the `AWSCURRENT` staging label.

```

data "aws_secretsmanager_secret_version" "example" {
  secret_id = "${data.aws_secretsmanager_secret.example.id}"
}

```



## » Retrieve Specific Secret Version

```
data "aws_secretsmanager_secret_version" "by-version-stage" {
  secret_id      = "${data.aws_secretsmanager_secret.example.id}"
  version_stage = "example"
}
```

## » Argument Reference

- **secret\_id** - (Required) Specifies the secret containing the version that you want to retrieve. You can specify either the Amazon Resource Name (ARN) or the friendly name of the secret.
- **version\_id** - (Optional) Specifies the unique identifier of the version of the secret that you want to retrieve. Overrides **version\_stage**.
- **version\_stage** - (Optional) Specifies the secret version that you want to retrieve by the staging label attached to the version. Defaults to **AWSCURRENT**.

## » Attributes Reference

- **id** - The unique identifier of this version of the secret.
- **secret\_string** - The decrypted part of the protected secret information that was originally provided as a string.
- **version\_id** - The unique identifier of this version of the secret.

## » Data Source: `aws__security__group`

`aws_security_group` provides details about a specific Security Group.

This resource can prove useful when a module accepts a Security Group id as an input variable and needs to, for example, determine the id of the VPC that the security group belongs to.

## » Example Usage

The following example shows how one might accept a Security Group id as a variable and use this data source to obtain the data necessary to create a subnet.

```
variable "security_group_id" {}

data "aws_security_group" "selected" {
  id = "${var.security_group_id}"
}
```

```

}

resource "aws_subnet" "subnet" {
  vpc_id      = "${data.aws_security_group.selected.vpc_id}"
  cidr_block = "10.0.1.0/24"
}

```

## » Argument Reference

The arguments of this data source act as filters for querying the available security group in the current region. The given filters must match exactly one security group whose data will be exported as attributes.

- **filter** - (Optional) Custom filter block as described below.
- **id** - (Optional) The id of the specific security group to retrieve.
- **name** - (Optional) The name that the desired security group must have.
- **tags** - (Optional) A mapping of tags, each pair of which must exactly match a pair on the desired security group.
- **vpc\_id** - (Optional) The id of the VPC that the desired security group belongs to.

More complex filters can be expressed using one or more **filter** sub-blocks, which take the following arguments:

- **name** - (Required) The name of the field to filter by, as defined by the underlying AWS API.
- **values** - (Required) Set of values that are accepted for the given field. A Security Group will be selected if any one of the given values matches.

## » Attributes Reference

All of the argument attributes except **filter** blocks are also exported as result attributes. This data source will complete the data by populating any fields that are not included in the configuration with the data for the selected Security Group.

The following fields are also exported:

- **description** - The description of the security group.
- **arn** - The computed ARN of the security group.

**Note:** The default security group for a VPC has the name **default**.

## » Data Source: `aws__sqs__queue`

Use this data source to get the ARN and URL of queue in AWS Simple Queue Service (SQS). By using this data source, you can reference SQS queues without having to hardcode the ARNs as input.

### » Example Usage

```
data "aws_sqs_queue" "example" {  
  name = "queue"  
}
```

### » Argument Reference

- `name` - (Required) The name of the queue to match.

### » Attributes Reference

- `arn` - The Amazon Resource Name (ARN) of the queue.
- `url` - The URL of the queue.

## » Data Source: `aws__sns__topic`

Use this data source to get the ARN of a topic in AWS Simple Notification Service (SNS). By using this data source, you can reference SNS topics without having to hard code the ARNs as input.

### » Example Usage

```
data "aws_sns_topic" "example" {  
  name = "an_example_topic"  
}
```

### » Argument Reference

- `name` - (Required) The friendly name of the topic to match.

## » Attributes Reference

- **arn** - Set to the ARN of the found topic, suitable for referencing in other resources that support SNS topics.

## » Data Source: `aws_ssm_parameter`

Provides an SSM Parameter data source.

## » Example Usage

To store a basic string parameter:

```
data "aws_ssm_parameter" "foo" {
  name = "foo"
}
```

**Note:** The unencrypted value of a `SecureString` will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the parameter.
- **with\_decryption** - (Optional) Whether to return decrypted `SecureString` value. Defaults to `true`.

The following attributes are exported:

- **arn** - The ARN of the parameter.
- **name** - (Required) The name of the parameter.
- **type** - (Required) The type of the parameter. Valid types are `String`, `StringList` and `SecureString`.
- **value** - (Required) The value of the parameter.

## » Data Source: `aws_subnet`

`aws_subnet` provides details about a specific VPC subnet.

This resource can prove useful when a module accepts a subnet id as an input variable and needs to, for example, determine the id of the VPC that the subnet belongs to.

## » Example Usage

The following example shows how one might accept a subnet id as a variable and use this data source to obtain the data necessary to create a security group that allows connections from hosts in that subnet.

```
variable "subnet_id" {}

data "aws_subnet" "selected" {
  id = "${var.subnet_id}"
}

resource "aws_security_group" "subnet" {
  vpc_id = "${data.aws_subnet.selected.vpc_id}"

  ingress {
    cidr_blocks = ["${data.aws_subnet.selected.cidr_block}"]
    from_port   = 80
    to_port     = 80
    protocol    = "tcp"
  }
}
```

## » Argument Reference

The arguments of this data source act as filters for querying the available subnets in the current region. The given filters must match exactly one subnet whose data will be exported as attributes.

- **availability\_zone** - (Optional) The availability zone where the subnet must reside.
- **cidr\_block** - (Optional) The cidr block of the desired subnet.
- **ipv6\_cidr\_block** - (Optional) The Ipv6 cidr block of the desired subnet
- **default\_for\_az** - (Optional) Boolean constraint for whether the desired subnet must be the default subnet for its associated availability zone.
- **filter** - (Optional) Custom filter block as described below.
- **id** - (Optional) The id of the specific subnet to retrieve.
- **state** - (Optional) The state that the desired subnet must have.
- **tags** - (Optional) A mapping of tags, each pair of which must exactly match a pair on the desired subnet.
- **vpc\_id** - (Optional) The id of the VPC that the desired subnet belongs to.

More complex filters can be expressed using one or more **filter** sub-blocks, which take the following arguments:

- **name** - (Required) The name of the field to filter by, as defined by the underlying AWS API. For example, if matching against tag **Name**, use:

```
filter {  
  name = "tag:Name"  
  values = ...  
}
```

- **values** - (Required) Set of values that are accepted for the given field. A subnet will be selected if any one of the given values matches.

## » Attributes Reference

All of the argument attributes except **filter** blocks are also exported as result attributes. This data source will complete the data by populating any fields that are not included in the configuration with the data for the selected subnet.

## » Data Source: `aws_subnet_ids`

`aws_subnet_ids` provides a list of ids for a `vpc_id`

This resource can be useful for getting back a list of subnet ids for a vpc.

## » Example Usage

The following shows outputting all cidr blocks for every subnet id in a vpc.

```
data "aws_subnet_ids" "example" {  
  vpc_id = "${var.vpc_id}"  
}  
  
data "aws_subnet" "example" {  
  count = "${length(data.aws_subnet_ids.example.ids)}"  
  id = "${data.aws_subnet_ids.example.ids[count.index]}"  
}  
  
output "subnet_cidr_blocks" {  
  value = ["${data.aws_subnet.example.*.cidr_block}"]  
}
```

The following example retrieves a list of all subnets in a VPC with a custom tag of `Tier` set to a value of "Private" so that the `aws_instance` resource can loop through the subnets, putting instances across availability zones.

```
data "aws_subnet_ids" "private" {
  vpc_id = "${var.vpc_id}"
  tags {
    Tier = "Private"
  }
}

resource "aws_instance" "app" {
  count          = "3"
  ami           = "${var.ami}"
  instance_type = "t2.micro"
  subnet_id     = "${element(data.aws_subnet_ids.private.ids, count.index)}"
}
```

## » Argument Reference

- `vpc_id` - (Required) The VPC ID that you want to filter from.
- `tags` - (Optional) A mapping of tags, each pair of which must exactly match a pair on the desired subnets.

## » Attributes Reference

- `ids` - A list of all the subnet ids found. This data source will fail if none are found.

## » Data Source: `aws_vpc`

`aws_vpc` provides details about a specific VPC.

This resource can prove useful when a module accepts a `vpc_id` as an input variable and needs to, for example, determine the CIDR block of that VPC.

## » Example Usage

The following example shows how one might accept a VPC id as a variable and use this data source to obtain the data necessary to create a subnet within it.

```

variable "vpc_id" {}

data "aws_vpc" "selected" {
  id = "${var.vpc_id}"
}

resource "aws_subnet" "example" {
  vpc_id            = "${data.aws_vpc.selected.id}"
  availability_zone = "us-west-2a"
  cidr_block        = "${cidrsubnet(data.aws_vpc.selected.cidr_block, 4, 1)}"
}

```

## » Argument Reference

The arguments of this data source act as filters for querying the available VPCs in the current region. The given filters must match exactly one VPC whose data will be exported as attributes.

- **cidr\_block** - (Optional) The cidr block of the desired VPC.
- **dhcp\_options\_id** - (Optional) The DHCP options id of the desired VPC.
- **default** - (Optional) Boolean constraint on whether the desired VPC is the default VPC for the region.
- **filter** - (Optional) Custom filter block as described below.
- **id** - (Optional) The id of the specific VPC to retrieve.
- **state** - (Optional) The current state of the desired VPC. Can be either "pending" or "available".
- **tags** - (Optional) A mapping of tags, each pair of which must exactly match a pair on the desired VPC.

More complex filters can be expressed using one or more **filter** sub-blocks, which take the following arguments:

- **name** - (Required) The name of the field to filter by, as defined by the underlying AWS API.
- **values** - (Required) Set of values that are accepted for the given field. A VPC will be selected if any one of the given values matches.

## » Attributes Reference

All of the argument attributes except **filter** blocks are also exported as result attributes. This data source will complete the data by populating any fields that are not included in the configuration with the data for the selected VPC.



The following attribute is additionally exported:

- `instance_tenancy` - The allowed tenancy of instances launched into the selected VPC. May be any of "default", "dedicated", or "host".
- `ipv6_association_id` - The association ID for the IPv6 CIDR block.
- `ipv6_cidr_block` - The IPv6 CIDR block.
- `enable_dns_support` - Whether or not the VPC has DNS support
- `enable_dns_hostnames` - Whether or not the VPC has DNS hostname support

## » Data Source: `aws_vpc_endpoint`

The VPC Endpoint data source provides details about a specific VPC endpoint.

### » Example Usage

```
# Declare the data source
data "aws_vpc_endpoint" "s3" {
  vpc_id      = "${aws_vpc.foo.id}"
  service_name = "com.amazonaws.us-west-2.s3"
}

resource "aws_vpc_endpoint_route_table_association" "private_s3" {
  vpc_endpoint_id = "${data.aws_vpc_endpoint.s3.id}"
  route_table_id  = "${aws_route_table.private.id}"
}
```

### » Argument Reference

The arguments of this data source act as filters for querying the available VPC endpoints. The given filters must match exactly one VPC endpoint whose data will be exported as attributes.

- `id` - (Optional) The ID of the specific VPC Endpoint to retrieve.
- `state` - (Optional) The state of the specific VPC Endpoint to retrieve.
- `vpc_id` - (Optional) The ID of the VPC in which the specific VPC Endpoint is used.
- `service_name` - (Optional) The AWS service name of the specific VPC Endpoint to retrieve.

## » Attributes Reference

All of the argument attributes are also exported as result attributes.

- **vpc\_endpoint\_type** - The VPC Endpoint type, **Gateway** or **Interface**.
- **policy** - The policy document associated with the VPC Endpoint. Applicable for endpoints of type **Gateway**.
- **route\_table\_ids** - One or more route tables associated with the VPC Endpoint. Applicable for endpoints of type **Gateway**.
- **prefix\_list\_id** - The prefix list ID of the exposed AWS service. Applicable for endpoints of type **Gateway**.
- **cidr\_blocks** - The list of CIDR blocks for the exposed AWS service. Applicable for endpoints of type **Gateway**.
- **subnet\_ids** - One or more subnets in which the VPC Endpoint is located. Applicable for endpoints of type **Interface**.
- **network\_interface\_ids** - One or more network interfaces for the VPC Endpoint. Applicable for endpoints of type **Interface**.
- **security\_group\_ids** - One or more security groups associated with the network interfaces. Applicable for endpoints of type **Interface**.
- **private\_dns\_enabled** - Whether or not the VPC is associated with a private hosted zone - **true** or **false**. Applicable for endpoints of type **Interface**.
- **dns\_entry** - The DNS entries for the VPC Endpoint. Applicable for endpoints of type **Interface**. DNS blocks are documented below.

DNS blocks (for **dns\_entry**) support the following attributes:

- **dns\_name** - The DNS name.
- **hosted\_zone\_id** - The ID of the private hosted zone.

## » Data Source: `aws_vpc_endpoint_service`

The VPC Endpoint Service data source details about a specific service that can be specified when creating a VPC endpoint within the region configured in the provider.

## » Example Usage

AWS service usage:

```
# Declare the data source
data "aws_vpc_endpoint_service" "s3" {
  service = "s3"
}
```

```

# Create a VPC
resource "aws_vpc" "foo" {
  cidr_block = "10.0.0.0/16"
}

# Create a VPC endpoint
resource "aws_vpc_endpoint" "ep" {
  vpc_id      = "${aws_vpc.foo.id}"
  service_name = "${data.aws_vpc_endpoint_service.s3.service_name}"
}

Non-AWS service usage:

data "aws_vpc_endpoint_service" "custome" {
  service_name = "com.amazonaws.vpce.us-west-2.vpce-svc-0e87519c997c63cd8"
}

```

## » Argument Reference

The arguments of this data source act as filters for querying the available VPC endpoint services. The given filters must match exactly one VPC endpoint service whose data will be exported as attributes.

- **service** - (Optional) The common name of an AWS service (e.g. `s3`).
- **service\_name** - (Optional) The service name that can be specified when creating a VPC endpoint.

**NOTE:** One of `service` or `service_name` must be specified.

## » Attributes Reference

The following attributes are exported:

- **service\_type** - The service type, `Gateway` or `Interface`.
- **owner** - The AWS account ID of the service owner or `amazon`.
- **vpce\_policy\_supported** - Whether or not the service supports endpoint policies - `true` or `false`.
- **acceptance\_required** - Whether or not VPC endpoint connection requests to the service must be accepted by the service owner - `true` or `false`.
- **availability\_zones** - The Availability Zones in which the service is available.
- **private\_dns\_name** - The private DNS name for the service.
- **base\_endpoint\_dns\_names** - The DNS names for the service.

## » Data Source: `aws_vpc_peering_connection`

The VPC Peering Connection data source provides details about a specific VPC peering connection.

### » Example Usage

```
# Declare the data source
data "aws_vpc_peering_connection" "pc" {
  vpc_id          = "${aws_vpc.foo.id}"
  peer_cidr_block = "10.0.1.0/22"
}

# Create a route table
resource "aws_route_table" "rt" {
  vpc_id = "${aws_vpc.foo.id}"
}

# Create a route
resource "aws_route" "r" {
  route_table_id      = "${aws_route_table.rt.id}"
  destination_cidr_block = "${data.aws_vpc_peering_connection.pc.peer_cidr_block}"
  vpc_peering_connection_id = "${data.aws_vpc_peering_connection.pc.id}"
}
```

### » Argument Reference

The arguments of this data source act as filters for querying the available VPC peering connection. The given filters must match exactly one VPC peering connection whose data will be exported as attributes.

- `id` - (Optional) The ID of the specific VPC Peering Connection to retrieve.
- `status` - (Optional) The status of the specific VPC Peering Connection to retrieve.
- `vpc_id` - (Optional) The ID of the requester VPC of the specific VPC Peering Connection to retrieve.
- `owner_id` - (Optional) The AWS account ID of the owner of the requester VPC of the specific VPC Peering Connection to retrieve.
- `cidr_block` - (Optional) The CIDR block of the requester VPC of the specific VPC Peering Connection to retrieve.
- `region` - (Optional) The region of the requester VPC of the specific VPC Peering Connection to retrieve.

- **peer\_vpc\_id** - (Optional) The ID of the acceptor VPC of the specific VPC Peering Connection to retrieve.
- **peer\_owner\_id** - (Optional) The AWS account ID of the owner of the acceptor VPC of the specific VPC Peering Connection to retrieve.
- **peer\_cidr\_block** - (Optional) The CIDR block of the acceptor VPC of the specific VPC Peering Connection to retrieve.
- **peer\_region** - (Optional) The region of the acceptor VPC of the specific VPC Peering Connection to retrieve.
- **filter** - (Optional) Custom filter block as described below.
- **tags** - (Optional) A mapping of tags, each pair of which must exactly match a pair on the desired VPC Peering Connection.

More complex filters can be expressed using one or more **filter** sub-blocks, which take the following arguments:

- **name** - (Required) The name of the field to filter by, as defined by the underlying AWS API.
- **values** - (Required) Set of values that are accepted for the given field. A VPC Peering Connection will be selected if any one of the given values matches.

## » Attributes Reference

All of the argument attributes except **filter** are also exported as result attributes.

- **accepter** - A configuration block that describes VPC Peering Connection options set for the acceptor VPC.
- **requester** - A configuration block that describes VPC Peering Connection options set for the requester VPC.

## » Acceptor and Requester Attributes Reference

- **allow\_remote\_vpc\_dns\_resolution** - Indicates whether a local VPC can resolve public DNS hostnames to private IP addresses when queried from instances in a peer VPC.
- **allow\_classic\_link\_to\_remote\_vpc** - Indicates whether a local ClassicLink connection can communicate with the peer VPC over the VPC peering connection.
- **allow\_vpc\_to\_remote\_classic\_link** - Indicates whether a local VPC can communicate with a ClassicLink connection in the peer VPC over the VPC peering connection.

## » Data Source: `aws__vpn__gateway`

The VPN Gateway data source provides details about a specific VPN gateway.

### » Example Usage

```
data "aws_vpn_gateway" "selected" {
  filter {
    name = "tag:Name"
    values = ["vpn-gw"]
  }
}

output "vpn_gateway_id" {
  value = "${data.aws_vpn_gateway.selected.id}"
}
```

### » Argument Reference

The arguments of this data source act as filters for querying the available VPN gateways. The given filters must match exactly one VPN gateway whose data will be exported as attributes.

- `id` - (Optional) The ID of the specific VPN Gateway to retrieve.
- `state` - (Optional) The state of the specific VPN Gateway to retrieve.
- `availability_zone` - (Optional) The Availability Zone of the specific VPN Gateway to retrieve.
- `attached_vpc_id` - (Optional) The ID of a VPC attached to the specific VPN Gateway to retrieve.
- `filter` - (Optional) Custom filter block as described below.
- `tags` - (Optional) A mapping of tags, each pair of which must exactly match a pair on the desired VPN Gateway.
- `amazon_side_asn` - (Optional) The Autonomous System Number (ASN) for the Amazon side of the specific VPN Gateway to retrieve.

More complex filters can be expressed using one or more `filter` sub-blocks, which take the following arguments:

- `name` - (Required) The name of the field to filter by, as defined by the underlying AWS API.
- `values` - (Required) Set of values that are accepted for the given field. A VPN Gateway will be selected if any one of the given values matches.

## » Attributes Reference

All of the argument attributes are also exported as result attributes.

## » `aws_acm_certificate`

The ACM certificate resource allows requesting and management of certificates from the Amazon Certificate Manager.

It deals with requesting certificates and managing their attributes and life-cycle. This resource does not deal with validation of a certificate but can provide inputs for other resources implementing the validation. It does not wait for a certificate to be issued. Use a `aws_acm_certificate_validation` resource for this.

Most commonly, this resource is used to together with `aws_route53_record` and `aws_acm_certificate_validation` to request a DNS validated certificate, deploy the required validation records and wait for validation to complete.

Domain validation through E-Mail is also supported but should be avoided as it requires a manual step outside of Terraform.

## » Example Usage

```
resource "aws_acm_certificate" "cert" {
  domain_name = "example.com"
  validation_method = "DNS"
  tags {
    Environment = "test"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `domain_name` - (Required) A domain name for which the certificate should be issued
- `subject_alternative_names` - (Optional) A list of domains that should be SANs in the issued certificate
- `validation_method` - (Required) Which method to use for validation. DNS or EMAIL are valid, NONE can be used for certificates that were imported into ACM and then into Terraform.
- `tags` - (Optional) A mapping of tags to assign to the resource.

## » Attributes Reference

The following additional attributes are exported:

- `id` - The ARN of the certificate
- `arn` - The ARN of the certificate
- `domain_validation_options` - A list of attributes to feed into other resources to complete certificate validation. Can have more than one element, e.g. if SANs are defined. Only set if `DNS`-validation was used.
- `validation_emails` - A list of addresses that received a validation E-Mail. Only set if `EMAIL`-validation was used.

Domain validation objects export the following attributes:

- `domain_name` - The domain to be validated
- `resource_record_name` - The name of the DNS record to create to validate the certificate
- `resource_record_type` - The type of DNS record to create
- `resource_record_value` - The value the DNS record needs to have

## » Import

Certificates can be imported using their ARN, e.g.

```
$ terraform import aws_acm_certificate.cert arn:aws:acm:eu-central-1:123456789012:certificate/12345678-1234-5678-9012-123456789012
```

**WARNING:** Importing certificates that are not `AMAZON_ISSUED` is supported but may lead to fragile terraform projects: Once such a resource is destroyed, it can't be recreated.

## » `aws__acm__certificate__validation`

This resource represents a successful validation of an ACM certificate in concert with other resources.

Most commonly, this resource is used to together with `aws_route53_record` and `aws_acm_certificate` to request a DNS validated certificate, deploy the required validation records and wait for validation to complete.

**WARNING:** This resource implements a part of the validation workflow. It does not represent a real-world entity in AWS, therefore changing or deleting this resource on its own has no immediate effect.



## » Example Usage

### » DNS Validation with Route 53

```
resource "aws_acm_certificate" "cert" {
  domain_name = "example.com"
  validation_method = "DNS"
}

data "aws_route53_zone" "zone" {
  name = "example.com."
  private_zone = false
}

resource "aws_route53_record" "cert_validation" {
  name = "${aws_acm_certificate.cert.domain_validation_options.0.resource_record_name}"
  type = "${aws_acm_certificate.cert.domain_validation_options.0.resource_record_type}"
  zone_id = "${data.aws_route53_zone.zone.id}"
  records = ["${aws_acm_certificate.cert.domain_validation_options.0.resource_record_value}"]
  ttl = 60
}

resource "aws_acm_certificate_validation" "cert" {
  certificate_arn = "${aws_acm_certificate.cert.arn}"
  validation_record_fqdns = ["${aws_route53_record.cert_validation.fqdn}"]
}

resource "aws_lb_listener" "front_end" {
  # [...]
  certificate_arn = "${aws_acm_certificate_validation.cert.certificate_arn}"
}
```

### » Email Validation

In this situation, the resource is simply a waiter for manual email approval of ACM certificates.

```
resource "aws_acm_certificate" "cert" {
  domain_name = "example.com"
  validation_method = "EMAIL"
}

resource "aws_acm_certificate_validation" "cert" {
  certificate_arn = "${aws_acm_certificate.cert.arn}"
}
```

## » Argument Reference

The following arguments are supported:

- **certificate\_arn** - (Required) The ARN of the certificate that is being validated.
- **validation\_record\_fqdns** - (Optional) List of FQDNs that implement the validation. Only valid for DNS validation method ACM certificates. If this is set, the resource can implement additional sanity checks and has an explicit dependency on the resource that is implementing the validation

## » Timeouts

**acm\_certificate\_validation** provides the following Timeouts configuration options:

- **create** - (Default 45m) How long to wait for a certificate to be issued.

## » aws\_api\_gateway\_account

Provides a settings of an API Gateway Account. Settings is applied region-wide per **provider** block.

**Note:** As there is no API method for deleting account settings or resetting it to defaults, destroying this resource will keep your account settings intact

## » Example Usage

```
resource "aws_api_gateway_account" "demo" {
  cloudwatch_role_arn = "${aws_iam_role.cloudwatch.arn}"
}

resource "aws_iam_role" "cloudwatch" {
  name = "api_gateway_cloudwatch_global"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "apigateway.amazonaws.com"
```

```

        },
        "Action": "sts:AssumeRole"
    }
]
}
EOF
}

resource "aws_iam_role_policy" "cloudwatch" {
    name = "default"
    role = "${aws_iam_role.cloudwatch.id}"

    policy = <<EOF
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:DescribeLogGroups",
                "logs:DescribeLogStreams",
                "logs:PutLogEvents",
                "logs:GetLogEvents",
                "logs:FilterLogEvents"
            ],
            "Resource": "*"
        }
    ]
}
EOF
}

```

## » Argument Reference

The following argument is supported:

- `cloudwatch_role_arn` - (Optional) The ARN of an IAM role for CloudWatch (to allow logging & monitoring). See more in AWS Docs. Logging & monitoring can be enabled/disabled and otherwise tuned on the API Gateway Stage level.

## » Attribute Reference

The following attribute is exported:

- `throttle_settings` - Account-Level throttle settings. See exported fields below.

`throttle_settings` block exports the following:

- `burst_limit` - The absolute maximum number of times API Gateway allows the API to be called per second (RPS).
- `rate_limit` - The number of times API Gateway allows the API to be called per second on average (RPS).

## » Import

API Gateway Accounts can be imported using the word `api-gateway-account`, e.g.

```
$ terraform import aws_api_gateway_account.demo api-gateway-account
```

## » `aws_api_gateway_api_key`

Provides an API Gateway API Key.

**Warning:** Since the API Gateway usage plans feature was launched on August 11, 2016, usage plans are now **required** to associate an API key with an API stage.

## » Example Usage

```
resource "aws_api_gateway_rest_api" "MyDemoAPI" {
  name = "MyDemoAPI"
}

resource "aws_api_gateway_api_key" "MyDemoApiKey" {
  name = "demo"

  stage_key {
    rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
    stage_name  = "${aws_api_gateway_deployment.MyDemoDeployment.stage_name}"
  }
}

resource "aws_api_gateway_deployment" "MyDemoDeployment" {
```

```

    rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
    stage_name   = "test"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the API key
- **description** - (Optional) The API key description. Defaults to "Managed by Terraform".
- **enabled** - (Optional) Specifies whether the API key can be used by callers. Defaults to **true**.
- **value** - (Optional) The value of the API key. If not specified, it will be automatically generated by AWS on creation.
- **stage\_key** - (Optional) A list of stage keys associated with the API key - see below

**stage\_key** block supports the following:

- **rest\_api\_id** - (Required) The ID of the associated REST API.
- **stage\_name** - (Required) The name of the API Gateway stage.

## » Attribute Reference

The following attributes are exported:

- **id** - The ID of the API key
- **created\_date** - The creation date of the API key
- **last\_updated\_date** - The last update date of the API key
- **value** - The value of the API key

## » Import

API Gateway Keys can be imported using the **id**, e.g.

```
$ terraform import aws_api_gateway_api_key.my_demo_key 8bklk8bl1k3sB38D9B310enyWT8c09B30lkq
```

## » aws\_\_api\_\_gateway\_\_authorizer

Provides an API Gateway Authorizer.

## » Example Usage

```
resource "aws_api_gateway_authorizer" "demo" {
  name                = "demo"
  rest_api_id         = "${aws_api_gateway_rest_api.demo.id}"
  authorizer_uri       = "${aws_lambda_function.authorizer.invoke_arn}"
  authorizer_credentials = "${aws_iam_role.invocation_role.arn}"
}

resource "aws_api_gateway_rest_api" "demo" {
  name = "auth-demo"
}

resource "aws_iam_role" "invocation_role" {
  name = "api_gateway_auth_invocation"
  path = "/"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "apigateway.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
EOF
}

resource "aws_iam_role_policy" "invocation_policy" {
  name = "default"
  role = "${aws_iam_role.invocation_role.id}"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "lambda:InvokeFunction",
      "Effect": "Allow",
```

```

        "Resource": "${aws_lambda_function.authorizer.arn}"
    }
]
}
EOF
}

resource "aws_iam_role" "lambda" {
    name = "demo-lambda"

    assume_role_policy = <<EOF
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "sts:AssumeRole",
            "Principal": {
                "Service": "lambda.amazonaws.com"
            },
            "Effect": "Allow",
            "Sid": ""
        }
    ]
}
EOF
}

resource "aws_lambda_function" "authorizer" {
    filename           = "lambda-function.zip"
    source_code_hash   = "${base64sha256(file("lambda-function.zip"))}"
    function_name      = "api_gateway_authorizer"
    role               = "${aws_iam_role.lambda.arn}"
    handler            = "exports.example"
}

```

## » Argument Reference

The following arguments are supported:

- **authorizer\_uri** - (Optional, required for type `TOKEN/REQUEST`) The authorizer's Uniform Resource Identifier (URI). This must be a well-formed Lambda function URI in the form of `arn:aws:apigateway:{region}:lambda:path/{service_api}`, e.g. `arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-west-2:123456789012:function:example`
- **name** - (Required) The name of the authorizer
- **rest\_api\_id** - (Required) The ID of the associated REST API

- `identity_source` - (Optional) The source of the identity in an incoming request. Defaults to `method.request.header.Authorization`. For `REQUEST` type, this may be a comma-separated list of values, including headers, query string parameters and stage variables - e.g. `"method.request.header.SomeHeaderName,method.request.querystring.SomeQueryStringName,stageVariables.SomeStageVariableName"`.
- `type` - (Optional) The type of the authorizer. Possible values are `TOKEN` for a Lambda function using a single authorization token submitted in a custom header, `REQUEST` for a Lambda function using incoming request parameters, or `COGNITO_USER_POOLS` for using an Amazon Cognito user pool. Defaults to `TOKEN`.
- `authorizer_credentials` - (Optional) The credentials required for the authorizer. To specify an IAM Role for API Gateway to assume, use the IAM Role ARN.
- `authorizer_result_ttl_in_seconds` - (Optional) The TTL of cached authorizer results in seconds. Defaults to 300.
- `identity_validation_expression` - (Optional) A validation expression for the incoming identity. For `TOKEN` type, this value should be a regular expression. The incoming token from the client is matched against this expression, and will proceed if the token matches. If the token doesn't match, the client receives a 401 Unauthorized response.
- `provider_arns` - (Optional, required for type `COGNITO_USER_POOLS`) A list of the Amazon Cognito user pool ARNs. Each element is of this format: `arn:aws:cognito-idp:{region}:{account_id}:userpool/{user_pool_id}`.

## » `aws_api_gateway_base_path_mapping`

Connects a custom domain name registered via `aws_api_gateway_domain_name` with a deployed API so that its methods can be called via the custom domain name.

### » Example Usage

```
resource "aws_api_gateway_deployment" "example" {
  # See aws_api_gateway_rest_api_docs for how to create this
  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  stage_name  = "live"
}

resource "aws_api_gateway_domain_name" "example" {
  domain_name = "example.com"

  certificate_name      = "example-api"
  certificate_body      = "${file("${path.module}/example.com/example.crt")}"
}
```



```

certificate_chain      = "${file("${path.module}/example.com/ca.crt")}"
certificate_private_key = "${file("${path.module}/example.com/example.key")}"
}

resource "aws_api_gateway_base_path_mapping" "test" {
  api_id      = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  stage_name  = "${aws_api_gateway_deployment.example.stage_name}"
  domain_name = "${aws_api_gateway_domain_name.example.domain_name}"
}

```

## » Argument Reference

The following arguments are supported:

- **domain\_name** - (Required) The already-registered domain name to connect the API to.
- **api\_id** - (Required) The id of the API to connect.
- **stage\_name** - (Optional) The name of a specific deployment stage to expose at the given path. If omitted, callers may select any stage by including its name as a path element after the base path.
- **base\_path** - (Optional) Path segment that must be prepended to the path when accessing the API via this mapping. If omitted, the API is exposed at the root of the given domain.

## » aws\_\_api\_\_gateway\_\_client\_\_certificate

Provides an API Gateway Client Certificate.

## » Example Usage

```

resource "aws_api_gateway_client_certificate" "demo" {
  description = "My client certificate"
}

```

## » Argument Reference

The following arguments are supported:

- **description** - (Optional) The description of the client certificate.

## » Attribute Reference

The following attributes are exported:

- `id` - The identifier of the client certificate.
- `created_date` - The date when the client certificate was created.
- `expiration_date` - The date when the client certificate will expire.
- `pem_encoded_certificate` - The PEM-encoded public key of the client certificate.

## » Import

API Gateway Client Certificates can be imported using the id, e.g.

```
$ terraform import aws_api_gateway_client_certificate.demo ab1cq
```

## » `aws_api_gateway_deployment`

Provides an API Gateway Deployment.

**Note:** Depends on having `aws_api_gateway_integration` inside your rest api (which in turn depends on `aws_api_gateway_method`). To avoid race conditions you might need to add an explicit `depends_on = ["aws_api_gateway_integration.name"]`.

## » Example Usage

```
resource "aws_api_gateway_rest_api" "MyDemoAPI" {
  name          = "MyDemoAPI"
  description   = "This is my API for demonstration purposes"
}

resource "aws_api_gateway_resource" "MyDemoResource" {
  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  parent_id   = "${aws_api_gateway_rest_api.MyDemoAPI.root_resource_id}"
  path_part   = "test"
}

resource "aws_api_gateway_method" "MyDemoMethod" {
  rest_api_id   = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  resource_id   = "${aws_api_gateway_resource.MyDemoResource.id}"
  http_method   = "GET"
  authorization = "NONE"
}
```

```

resource "aws_api_gateway_integration" "MyDemoIntegration" {
  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  resource_id = "${aws_api_gateway_resource.MyDemoResource.id}"
  http_method = "${aws_api_gateway_method.MyDemoMethod.http_method}"
  type        = "MOCK"
}

resource "aws_api_gateway_deployment" "MyDemoDeployment" {
  depends_on = ["aws_api_gateway_integration.MyDemoIntegration"]

  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  stage_name  = "test"

  variables = {
    "answer" = "42"
  }
}

```

## » Argument Reference

The following arguments are supported:

- `rest_api_id` - (Required) The ID of the associated REST API
- `stage_name` - (Required) The name of the stage
- `description` - (Optional) The description of the deployment
- `stage_description` - (Optional) The description of the stage
- `variables` - (Optional) A map that defines variables for the stage

## » Attribute Reference

The following attributes are exported:

- `id` - The ID of the deployment
- `invoke_url` - The URL to invoke the API pointing to the stage, e.g. `https://z4675bid1j.execute-api.eu-west-2.amazonaws.com/prod`
- `execution_arn` - The execution ARN to be used in `lambda_permission`'s `source_arn` when allowing API Gateway to invoke a Lambda function, e.g. `arn:aws:execute-api:eu-west-2:123456789012:z4675bid1j/prod`
- `created_date` - The creation date of the deployment

## » **aws\_api\_gateway\_documentation\_part**

Provides a settings of an API Gateway Documentation Part.

### » **Example Usage**

```
resource "aws_api_gateway_documentation_part" "example" {
  location {
    type = "METHOD"
    method = "GET"
    path = "/example"
  }
  properties = "{\"description\": \"Example description\"}"
  rest_api_id = "${aws_api_gateway_rest_api.example.id}"
}

resource "aws_api_gateway_rest_api" "example" {
  name = "example_api"
}
```

### » **Argument Reference**

The following argument is supported:

- **location** - (Required) The location of the targeted API entity of the to-be-created documentation part. See below.
- **properties** - (Required) A content map of API-specific key-value pairs describing the targeted API entity. The map must be encoded as a JSON string, e.g., "{ \"description\": \"The API does ...\" }". Only Swagger-compliant key-value pairs can be exported and, hence, published.
- **rest\_api\_id** - (Required) The ID of the associated Rest API

### » **Nested fields**

#### » **location**

See supported entity types for each field in the official docs.

- **method** - (Optional) The HTTP verb of a method. The default value is \* for any method.
- **name** - (Optional) The name of the targeted API entity.
- **path** - (Optional) The URL path of the target. The default value is / for the root resource.

- **status\_code** - (Optional) The HTTP status code of a response. The default value is \* for any status code.
- **type** - (Required) The type of API entity to which the documentation content applies. e.g. `API`, `METHOD` or `REQUEST_BODY`

## » Attribute Reference

The following attribute is exported in addition to the arguments listed above:

- **id** - The unique ID of the Documentation Part

## » Import

API Gateway `documentation_parts` can be imported using `REST-API-ID/DOC-PART-ID`, e.g.

```
$ terraform import aws_api_gateway_documentation_part.example 5i4e1ko720/3oyy3t
```

## » `aws_api_gateway_documentation_version`

Provides a resource to manage an API Gateway Documentation Version.

## » Example Usage

```
resource "aws_api_gateway_documentation_version" "example" {
  version = "example_version"
  rest_api_id = "${aws_api_gateway_rest_api.example.id}"
  description = "Example description"
  depends_on = ["aws_api_gateway_documentation_part.example"]
}

resource "aws_api_gateway_rest_api" "example" {
  name = "example_api"
}

resource "aws_api_gateway_documentation_part" "example" {
  location {
    type = "API"
  }
  properties = "{\"description\": \"Example\"}"
  rest_api_id = "${aws_api_gateway_rest_api.example.id}"
}
```

## » Argument Reference

The following argument is supported:

- **version** - (Required) The version identifier of the API documentation snapshot.
- **rest\_api\_id** - (Required) The ID of the associated Rest API
- **description** - (Optional) The description of the API documentation version.

## » Attribute Reference

The arguments listed above are all exported as attributes.

## » Import

API Gateway documentation versions can be imported using `REST-API-ID/VERSION`, e.g.

```
$ terraform import aws_api_gateway_documentation_version.example 5i4e1ko720/example-version
```

## » `aws_api_gateway_domain_name`

Registers a custom domain name for use with AWS API Gateway.

This resource just establishes ownership of and the TLS settings for a particular domain name. An API can be attached to a particular path under the registered domain name using the `aws_api_gateway_base_path_mapping` resource.

Internally API Gateway creates a CloudFront distribution to route requests on the given hostname. In addition to this resource it's necessary to create a DNS record corresponding to the given domain name which is an alias (either Route53 alias or traditional CNAME) to the Cloudfront domain name exported in the `cloudfront_domain_name` attribute.

**Note:** All arguments including the private key will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Example Usage

```
resource "aws_api_gateway_domain_name" "example" {
  domain_name = "api.example.com"

  certificate_name = "example-api"
```

```

certificate_body      = "${file("${path.module}/example.com/example.crt")}"
certificate_chain     = "${file("${path.module}/example.com/ca.crt")}"
certificate_private_key = "${file("${path.module}/example.com/example.key")}"
}

# Example DNS record using Route53.
# Route53 is not specifically required; any DNS host can be used.
resource "aws_route53_record" "example" {
  zone_id = "${aws_route53_zone.example.id}" # See aws_route53_zone for how to create this

  name = "${aws_api_gateway_domain_name.example.domain_name}"
  type = "A"

  alias {
    name          = "${aws_api_gateway_domain_name.example.cloudfront_domain_name}"
    zone_id       = "${aws_api_gateway_domain_name.example.cloudfront_zone_id}"
    evaluate_target_health = true
  }
}

```

## » Argument Reference

The following arguments are supported:

- **domain\_name** - (Required) The fully-qualified domain name to register
- **certificate\_name** - (Optional) The unique name to use when registering this cert as an IAM server certificate. Conflicts with **certificate\_arn**. Required if **certificate\_arn** is not set.
- **certificate\_body** - (Optional) The certificate issued for the domain name being registered, in PEM format. Conflicts with **certificate\_arn**.
- **certificate\_chain** - (Optional) The certificate for the CA that issued the certificate, along with any intermediate CA certificates required to create an unbroken chain to a certificate trusted by the intended API clients. Conflicts with **certificate\_arn**.
- **certificate\_private\_key** - (Optional) The private key associated with the domain certificate given in **certificate\_body**. Conflicts with **certificate\_arn**.
- **certificate\_arn** - (Optional) The ARN for an AWS-managed certificate. Conflicts with **certificate\_name**, **certificate\_body**, **certificate\_chain** and **certificate\_private\_key**.

## » Attributes Reference

In addition to the arguments, the following attributes are exported:

- **id** - The internal id assigned to this domain name by API Gateway.
- **certificate\_upload\_date** - The upload date associated with the domain certificate.
- **cloudfront\_domain\_name** - The hostname created by Cloudfront to represent the distribution that implements this domain name mapping.
- **cloudfront\_zone\_id** - For convenience, the hosted zone id (Z2FDTNDATAQYW2) that can be used to create a Route53 alias record for the distribution.

## » **aws\_api\_gateway\_gateway\_response**

Provides an API Gateway Gateway Response for a REST API Gateway.

### » **Example Usage**

```
resource "aws_api_gateway_rest_api" "main" {
  name = "MyDemoAPI"
}

resource "aws_api_gateway_gateway_response" "test" {
  rest_api_id   = "${aws_api_gateway_rest_api.main.id}"
  status_code   = "401"
  response_type = "UNAUTHORIZED"

  response_templates = {
    "application/json" = "{\"message\":$context.error.messageString}"
  }

  response_parameters = {
    "gatewayresponse.header.Authorization" = "'Basic'"
  }
}
```

### » **Argument Reference**

The following arguments are supported:

- **rest\_api\_id** - (Required) The string identifier of the associated REST API.
- **response\_type** - (Required) The response type of the associated GatewayResponse.
- **status\_code** - (Optional) The HTTP status code of the Gateway Response.



- `response_parameters` - (Optional) A map specifying the templates used to transform the response body.
- `response_templates` - (Optional) A map specifying the parameters (paths, query strings and headers) of the Gateway Response.

## » `aws__api_gateway__integration`

Provides an HTTP Method Integration for an API Gateway Integration.

### » Example Usage

```
resource "aws_api_gateway_rest_api" "MyDemoAPI" {
  name          = "MyDemoAPI"
  description   = "This is my API for demonstration purposes"
}

resource "aws_api_gateway_resource" "MyDemoResource" {
  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  parent_id   = "${aws_api_gateway_rest_api.MyDemoAPI.root_resource_id}"
  path_part   = "mydemoresource"
}

resource "aws_api_gateway_method" "MyDemoMethod" {
  rest_api_id   = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  resource_id   = "${aws_api_gateway_resource.MyDemoResource.id}"
  http_method   = "GET"
  authorization = "NONE"
}

resource "aws_api_gateway_integration" "MyDemoIntegration" {
  rest_api_id      = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  resource_id      = "${aws_api_gateway_resource.MyDemoResource.id}"
  http_method      = "${aws_api_gateway_method.MyDemoMethod.http_method}"
  type             = "MOCK"
  cache_key_parameters = ["method.request.path.param"]
  cache_namespace  = "foobar"

  request_parameters = {
    "integration.request.header.X-Authorization" = "'static'"
  }

  # Transforms the incoming XML request to JSON
  request_templates {
```

```

    "application/xml" = <<EOF
  {
    "body" : $input.json('$')
  }
  EOF
}
}

```

## » Lambda integration

```

# Variables
variable "myregion" {}
variable "accountId" {}

# API Gateway
resource "aws_api_gateway_rest_api" "api" {
  name = "myapi"
}

resource "aws_api_gateway_resource" "resource" {
  path_part = "resource"
  parent_id = "${aws_api_gateway_rest_api.api.root_resource_id}"
  rest_api_id = "${aws_api_gateway_rest_api.api.id}"
}

resource "aws_api_gateway_method" "method" {
  rest_api_id   = "${aws_api_gateway_rest_api.api.id}"
  resource_id   = "${aws_api_gateway_resource.resource.id}"
  http_method   = "GET"
  authorization = "NONE"
}

resource "aws_api_gateway_integration" "integration" {
  rest_api_id           = "${aws_api_gateway_rest_api.api.id}"
  resource_id           = "${aws_api_gateway_resource.resource.id}"
  http_method           = "${aws_api_gateway_method.method.http_method}"
  integration_http_method = "POST"
  type                  = "AWS_PROXY"
  uri                   = "arn:aws:apigateway:${var.myregion}:lambda:path/2015-03-31/functions/"
}

# Lambda
resource "aws_lambda_permission" "apigw_lambda" {
  statement_id = "AllowExecutionFromAPIGateway"
  action       = "lambda:InvokeFunction"
}

```

```

function_name = "${aws_lambda_function.lambda.arn}"
principal      = "apigateway.amazonaws.com"

# More: http://docs.aws.amazon.com/apigateway/latest/developerguide/api-gateway-control-a
source_arn = "arn:aws:execute-api:${var.myregion}:${var.accountId}:${aws_api_gateway_rest.
}

resource "aws_lambda_function" "lambda" {
  filename      = "lambda.zip"
  function_name = "mylambda"
  role          = "${aws_iam_role.role.arn}"
  handler       = "lambda.lambda_handler"
  runtime       = "python2.7"
  source_code_hash = "${base64sha256(file("lambda.zip"))}"
}

# IAM
resource "aws_iam_role" "role" {
  name = "myrole"

  assume_role_policy = <<POLICY
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "lambda.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
POLICY
}

```

## » VPC Link

```

variable "name" {}
variable "subnet_id" {}

resource "aws_lb" "test" {
  name          = "${var.name}"
  internal      = true

```

```

    load_balancer_type = "network"
    subnets            = ["${var.subnet_id}"]
}

resource "aws_api_gateway_vpc_link" "test" {
  name      = "${var.name}"
  target_arns = ["${aws_lb.test.arn}"]
}

resource "aws_api_gateway_rest_api" "test" {
  name = "${var.name}"
}

resource "aws_api_gateway_resource" "test" {
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  parent_id   = "${aws_api_gateway_rest_api.test.root_resource_id}"
  path_part   = "test"
}

resource "aws_api_gateway_method" "test" {
  rest_api_id   = "${aws_api_gateway_rest_api.test.id}"
  resource_id   = "${aws_api_gateway_resource.test.id}"
  http_method   = "GET"
  authorization = "NONE"

  request_models = {
    "application/json" = "Error"
  }
}

resource "aws_api_gateway_integration" "test" {
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  resource_id = "${aws_api_gateway_resource.test.id}"
  http_method = "${aws_api_gateway_method.test.http_method}"

  request_templates = {
    "application/json" = ""
    "application/xml"  = "#set($inputRoot = $input.path('$'))\n{ }"
  }

  request_parameters = {
    "integration.request.header.X-Authorization" = "'static'"
    "integration.request.header.X-Foo"           = "'Bar'"
  }

  type = "HTTP"

```



the caller's identity be passed through from the request, specify the string `arn:aws:iam::\*:user/\*`.

- `request_templates` - (Optional) A map of the integration's request templates.
- `request_parameters` - (Optional) A map of request query string parameters and headers that should be passed to the backend responder. For example: `request_parameters = { "integration.request.header.X-Some-Other-Header" = "method.request.header.X-Some-Header" }`
- `passthrough_behavior` - (Optional) The integration passthrough behavior (`WHEN_NO_MATCH`, `WHEN_NO_TEMPLATES`, `NEVER`). **Required** if `request_templates` is used.
- `cache_key_parameters` - (Optional) A list of cache key parameters for the integration.
- `cache_namespace` - (Optional) The integration's cache namespace.
- `request_parameters_in_json` - **Deprecated**, use `request_parameters` instead.
- `content_handling` - (Optional) Specifies how to handle request payload content type conversions. Supported values are `CONVERT_TO_BINARY` and `CONVERT_TO_TEXT`. If this property is not defined, the request payload will be passed through from the method request to integration request without modification, provided that the `passthroughBehaviors` is configured to support payload pass-through.

## » `aws_api_gateway_integration_response`

Provides an HTTP Method Integration Response for an API Gateway Resource.

**Note:** Depends on having `aws_api_gateway_integration` inside your rest api. To ensure this you might need to add an explicit `depends_on` for clean runs.

### » Example Usage

```
resource "aws_api_gateway_rest_api" "MyDemoAPI" {
  name          = "MyDemoAPI"
  description   = "This is my API for demonstration purposes"
}

resource "aws_api_gateway_resource" "MyDemoResource" {
  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  parent_id   = "${aws_api_gateway_rest_api.MyDemoAPI.root_resource_id}"
  path_part   = "mydemoresource"
}

resource "aws_api_gateway_method" "MyDemoMethod" {
```

```

    rest_api_id    = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
    resource_id    = "${aws_api_gateway_resource.MyDemoResource.id}"
    http_method    = "GET"
    authorization  = "NONE"
}

resource "aws_api_gateway_integration" "MyDemoIntegration" {
    rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
    resource_id = "${aws_api_gateway_resource.MyDemoResource.id}"
    http_method = "${aws_api_gateway_method.MyDemoMethod.http_method}"
    type        = "MOCK"
}

resource "aws_api_gateway_method_response" "200" {
    rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
    resource_id = "${aws_api_gateway_resource.MyDemoResource.id}"
    http_method = "${aws_api_gateway_method.MyDemoMethod.http_method}"
    status_code = "200"
}

resource "aws_api_gateway_integration_response" "MyDemoIntegrationResponse" {
    rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
    resource_id = "${aws_api_gateway_resource.MyDemoResource.id}"
    http_method = "${aws_api_gateway_method.MyDemoMethod.http_method}"
    status_code = "${aws_api_gateway_method_response.200.status_code}"

    # Transforms the backend JSON response to XML
    response_templates {
        "application/xml" = <<EOF
#set($inputRoot = $input.path('$'))
<?xml version="1.0" encoding="UTF-8"?>
<message>
    $inputRoot.body
</message>
EOF
    }
}

```

## » Argument Reference

The following arguments are supported:

- `rest_api_id` - (Required) The ID of the associated REST API
- `resource_id` - (Required) The API resource ID
- `http_method` - (Required) The HTTP method (GET, POST, PUT, DELETE,

HEAD, OPTIONS, ANY)

- **status\_code** - (Required) The HTTP status code
- **selection\_pattern** - (Optional) Specifies the regular expression pattern used to choose an integration response based on the response from the backend. Setting this to - makes the integration the default one. If the backend is an AWS Lambda function, the AWS Lambda function error header is matched. For all other HTTP and AWS backends, the HTTP status code is matched.
- **response\_templates** - (Optional) A map specifying the templates used to transform the integration response body
- **response\_parameters** - (Optional) A map of response parameters that can be read from the backend response. For example: `response_parameters = { "method.response.header.X-Some-Header" = "integration.response.header.X-Some-Other-Header" },`
- **response\_parameters\_in\_json** - **Deprecated**, use **response\_parameters** instead.
- **content\_handling** - (Optional) Specifies how to handle request payload content type conversions. Supported values are `CONVERT_TO_BINARY` and `CONVERT_TO_TEXT`. If this property is not defined, the response payload will be passed through from the integration response to the method response without modification.

## » `aws_api_gateway_method`

Provides a HTTP Method for an API Gateway Resource.

### » Example Usage

```
resource "aws_api_gateway_rest_api" "MyDemoAPI" {
  name          = "MyDemoAPI"
  description    = "This is my API for demonstration purposes"
}

resource "aws_api_gateway_resource" "MyDemoResource" {
  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  parent_id   = "${aws_api_gateway_rest_api.MyDemoAPI.root_resource_id}"
  path_part   = "mydemoresource"
}

resource "aws_api_gateway_method" "MyDemoMethod" {
  rest_api_id    = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  resource_id     = "${aws_api_gateway_resource.MyDemoResource.id}"
  http_method    = "GET"
```



```

    authorization = "NONE"
}

```

## » Usage with Cognito User Pool Authorizer

```

variable "cognito_user_pool_name" {}

data "aws_cognito_user_pools" "this" {
  name = "${var.cognito_user_pool_name}"
}

resource "aws_api_gateway_rest_api" "this" {
  name = "with-authorizer"
}

resource "aws_api_gateway_resource" "this" {
  rest_api_id = "${aws_api_gateway_rest_api.this.id}"
  parent_id   = "${aws_api_gateway_rest_api.this.root_resource_id}"
  path_part   = "{proxy+}"
}

resource "aws_api_gateway_authorizer" "this" {
  name          = "CognitoUserPoolAuthorizer"
  type          = "COGNITO_USER_POOLS"
  rest_api_id   = "${aws_api_gateway_rest_api.this.id}"
  provider_arns = ["${data.aws_cognito_user_pools.this.arns}"]
}

resource "aws_api_gateway_method" "any" {
  rest_api_id   = "${aws_api_gateway_rest_api.this.id}"
  resource_id   = "${aws_api_gateway_resource.this.id}"
  http_method   = "ANY"
  authorization = "COGNITO_USER_POOLS"
  authorizer_id = "${aws_api_gateway_authorizer.this.id}"

  request_parameters = {
    "method.request.path.proxy" = true
  }
}

```

## » Argument Reference

The following arguments are supported:

- `rest_api_id` - (Required) The ID of the associated REST API
- `resource_id` - (Required) The API resource ID
- `http_method` - (Required) The HTTP Method (GET, POST, PUT, DELETE, HEAD, OPTIONS, ANY)
- `authorization` - (Required) The type of authorization used for the method (NONE, CUSTOM, AWS\_IAM)
- `authorizer_id` - (Optional) The authorizer id to be used when the authorization is CUSTOM
- `api_key_required` - (Optional) Specify if the method requires an API key
- `request_models` - (Optional) A map of the API models used for the request's content type where key is the content type (e.g. `application/json`) and value is either `Error`, `Empty` (built-in models) or `aws_api_gateway_model`'s name.
- `request_validator_id` - (Optional) The ID of a `aws_api_gateway_request_validator`
- `request_parameters` - (Optional) A map of request query string parameters and headers that should be passed to the integration. For example: `hcl request_parameters = { "method.request.header.X-Some-Header" = true, "method.request.querystring.some-query-param" = true, }` would define that the header `X-Some-Header` and the query string `some-query-param` must be provided on the request, or
- `request_parameters_in_json` - **Deprecated**, use `request_parameters` instead.

## » `aws_api_gateway_method_response`

Provides an HTTP Method Response for an API Gateway Resource.

### » Example Usage

```
resource "aws_api_gateway_rest_api" "MyDemoAPI" {
  name          = "MyDemoAPI"
  description   = "This is my API for demonstration purposes"
}

resource "aws_api_gateway_resource" "MyDemoResource" {
  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  parent_id   = "${aws_api_gateway_rest_api.MyDemoAPI.root_resource_id}"
  path_part   = "mydemoresource"
}

resource "aws_api_gateway_method" "MyDemoMethod" {
  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
```

```

    resource_id    = "${aws_api_gateway_resource.MyDemoResource.id}"
    http_method    = "GET"
    authorization  = "NONE"
}

resource "aws_api_gateway_integration" "MyDemoIntegration" {
    rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
    resource_id = "${aws_api_gateway_resource.MyDemoResource.id}"
    http_method = "${aws_api_gateway_method.MyDemoMethod.http_method}"
    type        = "MOCK"
}

resource "aws_api_gateway_method_response" "200" {
    rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
    resource_id = "${aws_api_gateway_resource.MyDemoResource.id}"
    http_method = "${aws_api_gateway_method.MyDemoMethod.http_method}"
    status_code = "200"
}

```

## » Argument Reference

The following arguments are supported:

- `rest_api_id` - (Required) The ID of the associated REST API
- `resource_id` - (Required) The API resource ID
- `http_method` - (Required) The HTTP Method (GET, POST, PUT, DELETE, HEAD, OPTIONS, ANY)
- `status_code` - (Required) The HTTP status code
- `response_models` - (Optional) A map of the API models used for the response's content type
- `response_parameters` - (Optional) A map of response parameters that can be sent to the caller. For example: `response_parameters = { "method.response.header.X-Some-Header" = true }` would define that the header X-Some-Header can be provided on the response.
- `response_parameters_in_json` - **Deprecated**, use `response_parameters` instead.

## » `aws__api_gateway__method__settings`

Provides an API Gateway Method Settings, e.g. logging or monitoring.

## » Example Usage

```
resource "aws_api_gateway_method_settings" "s" {
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  stage_name  = "${aws_api_gateway_stage.test.stage_name}"
  method_path = "${aws_api_gateway_resource.test.path_part}/${aws_api_gateway_method.test.ht

  settings {
    metrics_enabled = true
    logging_level   = "INFO"
  }
}

resource "aws_api_gateway_rest_api" "test" {
  name = "MyDemoAPI"
  description = "This is my API for demonstration purposes"
}

resource "aws_api_gateway_deployment" "test" {
  depends_on = ["aws_api_gateway_integration.test"]
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  stage_name = "dev"
}

resource "aws_api_gateway_stage" "test" {
  stage_name = "prod"
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  deployment_id = "${aws_api_gateway_deployment.test.id}"
}

resource "aws_api_gateway_resource" "test" {
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  parent_id   = "${aws_api_gateway_rest_api.test.root_resource_id}"
  path_part   = "mytestresource"
}

resource "aws_api_gateway_method" "test" {
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  resource_id = "${aws_api_gateway_resource.test.id}"
  http_method = "GET"
  authorization = "NONE"
}

resource "aws_api_gateway_integration" "test" {
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
```

```

resource_id = "${aws_api_gateway_resource.test.id}"
http_method = "${aws_api_gateway_method.test.http_method}"
type        = "MOCK"

request_templates {
  "application/xml" = <<EOF
{
  "body" : $input.json('$')
}
EOF
}
}

```

## » Argument Reference

The following arguments are supported:

- **rest\_api\_id** - (Required) The ID of the REST API
- **stage\_name** - (Required) The name of the stage
- **method\_path** - (Required) Method path defined as `{resource_path}/{http_method}` for an individual method override, or `*/*` for overriding all methods in the stage.
- **settings** - (Required) The settings block, see below.

### » settings

- **metrics\_enabled** - (Optional) Specifies whether Amazon CloudWatch metrics are enabled for this method.
- **logging\_level** - (Optional) Specifies the logging level for this method, which effects the log entries pushed to Amazon CloudWatch Logs. The available levels are **OFF**, **ERROR**, and **INFO**.
- **data\_trace\_enabled** - (Optional) Specifies whether data trace logging is enabled for this method, which effects the log entries pushed to Amazon CloudWatch Logs.
- **throttling\_burst\_limit** - (Optional) Specifies the throttling burst limit.
- **throttling\_rate\_limit** - (Optional) Specifies the throttling rate limit.
- **caching\_enabled** - (Optional) Specifies whether responses should be cached and returned for requests. A cache cluster must be enabled on the stage for responses to be cached.
- **cache\_ttl\_in\_seconds** - (Optional) Specifies the time to live (TTL), in seconds, for cached responses. The higher the TTL, the longer the response will be cached.
- **cache\_data\_encrypted** - (Optional) Specifies whether the cached responses are encrypted.

- `require_authorization_for_cache_control` - (Optional) Specifies whether authorization is required for a cache invalidation request.
- `unauthorized_cache_control_header_strategy` - (Optional) Specifies how to handle unauthorized requests for cache invalidation. The available values are `FAIL_WITH_403`, `SUCCEED_WITH_RESPONSE_HEADER`, `SUCCEED_WITHOUT_RESPONSE_HEADER`.

## » `aws__api_gateway__model`

Provides a Model for a API Gateway.

### » Example Usage

```
resource "aws_api_gateway_rest_api" "MyDemoAPI" {
  name          = "MyDemoAPI"
  description   = "This is my API for demonstration purposes"
}

resource "aws_api_gateway_model" "MyDemoModel" {
  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  name        = "user"
  description = "a JSON schema"
  content_type = "application/json"

  schema = <<EOF
{
  "type": "object"
}
EOF
}
```

### » Argument Reference

The following arguments are supported:

- `rest_api_id` - (Required) The ID of the associated REST API
- `name` - (Required) The name of the model
- `description` - (Optional) The description of the model
- `content_type` - (Required) The content type of the model
- `schema` - (Required) The schema of the model in a JSON form

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the model

## » `aws_api_gateway_resource`

Provides an API Gateway Resource.

## » Example Usage

```
resource "aws_api_gateway_rest_api" "MyDemoAPI" {
  name      = "MyDemoAPI"
  description = "This is my API for demonstration purposes"
}

resource "aws_api_gateway_resource" "MyDemoResource" {
  rest_api_id = "${aws_api_gateway_rest_api.MyDemoAPI.id}"
  parent_id   = "${aws_api_gateway_rest_api.MyDemoAPI.root_resource_id}"
  path_part   = "mydemoresource"
}
```

## » Argument Reference

The following arguments are supported:

- `rest_api_id` - (Required) The ID of the associated REST API
- `parent_id` - (Required) The ID of the parent API resource
- `path_part` - (Required) The last path segment of this API resource.

## » Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `id` - The resource's identifier.
- `path` - The complete path for this API resource, including all parent paths.

## » `aws_api_gateway_rest_api`

Provides an API Gateway REST API.

## » Example Usage

```
resource "aws_api_gateway_rest_api" "MyDemoAPI" {  
  name      = "MyDemoAPI"  
  description = "This is my API for demonstration purposes"  
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the REST API
- **description** - (Optional) The description of the REST API
- **binary\_media\_types** - (Optional) The list of binary media types supported by the RestApi. By default, the RestApi supports only UTF-8-encoded text payloads.
- **minimum\_compression\_size** - (Optional) Minimum response size to compress for the REST API. Integer between -1 and 10485760 (10MB). Setting a value greater than -1 will enable compression, -1 disables compression (default).
- **body** - (Optional) An OpenAPI specification that defines the set of routes and integrations to create as part of the REST API.
- **policy** - (Optional) JSON formatted policy document that controls access to the API Gateway

**Note:** If the **body** argument is provided, the OpenAPI specification will be used to configure the resources, methods and integrations for the Rest API. If this argument is provided, the following resources should not be managed as separate ones, as updates may cause manual resource updates to be overwritten:

- `aws_api_gateway_resource`
- `aws_api_gateway_method`
- `aws_api_gateway_method_response`
- `aws_api_gateway_method_settings`
- `aws_api_gateway_integration`
- `aws_api_gateway_integration_response`
- `aws_api_gateway_gateway_response`
- `aws_api_gateway_model`

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the REST API
- **root\_resource\_id** - The resource ID of the REST API's root
- **created\_date** - The creation date of the REST API



## » `aws_api_gateway_stage`

Provides an API Gateway Stage.

### » Example Usage

```
resource "aws_api_gateway_stage" "test" {
  stage_name = "prod"
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  deployment_id = "${aws_api_gateway_deployment.test.id}"
}

resource "aws_api_gateway_rest_api" "test" {
  name = "MyDemoAPI"
  description = "This is my API for demonstration purposes"
}

resource "aws_api_gateway_deployment" "test" {
  depends_on = ["aws_api_gateway_integration.test"]
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  stage_name = "dev"
}

resource "aws_api_gateway_resource" "test" {
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  parent_id   = "${aws_api_gateway_rest_api.test.root_resource_id}"
  path_part   = "mytestresource"
}

resource "aws_api_gateway_method" "test" {
  rest_api_id   = "${aws_api_gateway_rest_api.test.id}"
  resource_id    = "${aws_api_gateway_resource.test.id}"
  http_method    = "GET"
  authorization   = "NONE"
}

resource "aws_api_gateway_method_settings" "s" {
  rest_api_id = "${aws_api_gateway_rest_api.test.id}"
  stage_name  = "${aws_api_gateway_stage.test.stage_name}"
  method_path = "${aws_api_gateway_resource.test.path_part}/${aws_api_gateway_method.test.http_method}"

  settings {
    metrics_enabled = true
    logging_level   = "INFO"
  }
}
```

```

    }
  }

  resource "aws_api_gateway_integration" "test" {
    rest_api_id = "${aws_api_gateway_rest_api.test.id}"
    resource_id = "${aws_api_gateway_resource.test.id}"
    http_method = "${aws_api_gateway_method.test.http_method}"
    type        = "MOCK"
  }

```

## » Argument Reference

The following arguments are supported:

- **rest\_api\_id** - (Required) The ID of the associated REST API
- **stage\_name** - (Required) The name of the stage
- **deployment\_id** - (Required) The ID of the deployment that the stage points to
- **access\_log\_settings** - (Optional) Enables access logs for the API stage. Detailed below.
- **cache\_cluster\_enabled** - (Optional) Specifies whether a cache cluster is enabled for the stage
- **cache\_cluster\_size** - (Optional) The size of the cache cluster for the stage, if enabled. Allowed values include 0.5, 1.6, 6.1, 13.5, 28.4, 58.2, 118 and 237.
- **client\_certificate\_id** - (Optional) The identifier of a client certificate for the stage.
- **description** - (Optional) The description of the stage
- **documentation\_version** - (Optional) The version of the associated API documentation
- **variables** - (Optional) A map that defines the stage variables
- **tags** - (Optional) A mapping of tags to assign to the resource.

## » Nested Blocks

### » **access\_log\_settings**

- **destination\_arn** - (Required) ARN of the log group to send the logs to. Automatically removes trailing `:*` if present.
- **format** - (Required) The formatting and values recorded in the logs. For more information on configuring the log format rules visit the AWS documentation

## » Attribute Reference

The following attributes are exported:

- `id` - The ID of the stage
- `invoke_url` - The URL to invoke the API pointing to the stage, e.g. `https://z4675bid1j.execute-api.eu-west-2.amazonaws.com/prod`
- `execution_arn` - The execution ARN to be used in `lambda_permission`'s `source_arn` when allowing API Gateway to invoke a Lambda function, e.g. `arn:aws:execute-api:eu-west-2:123456789012:z4675bid1j/prod`

## » `aws_api_gateway_usage_plan`

Provides an API Gateway Usage Plan.

### » Example Usage

```
resource "aws_api_gateway_rest_api" "myapi" {
  name = "MyDemoAPI"
}

...

resource "aws_api_gateway_deployment" "dev" {
  rest_api_id = "${aws_api_gateway_rest_api.myapi.id}"
  stage_name = "dev"
}

resource "aws_api_gateway_deployment" "prod" {
  rest_api_id = "${aws_api_gateway_rest_api.myapi.id}"
  stage_name = "prod"
}

resource "aws_api_gateway_usage_plan" "MyUsagePlan" {
  name          = "my-usage-plan"
  description   = "my description"
  product_code = "MYCODE"

  api_stages {
    api_id = "${aws_api_gateway_rest_api.myapi.id}"
    stage  = "${aws_api_gateway_deployment.dev.stage_name}"
  }

  api_stages {
```

```

    api_id = "${aws_api_gateway_rest_api.myapi.id}"
    stage = "${aws_api_gateway_deployment.prod.stage_name}"
}

quota_settings {
    limit = 20
    offset = 2
    period = "WEEK"
}

throttle_settings {
    burst_limit = 5
    rate_limit = 10
}
}

```

## » Argument Reference

The API Gateway Usage Plan argument layout is a structure composed of several sub-resources - these resources are laid out below.

### » Top-Level Arguments

- **name** - (Required) The name of the usage plan.
- **description** - (Optional) The description of a usage plan.
- **api\_stages** - (Optional) The associated API stages of the usage plan.
- **quota\_settings** - (Optional) The quota settings of the usage plan.
- **throttle\_settings** - (Optional) The throttling limits of the usage plan.
- **product\_code** - (Optional) The AWS Marketplace product identifier to associate with the usage plan as a SaaS product on AWS Marketplace.

### » Api Stages arguments

- **api\_id** (Required) - API Id of the associated API stage in a usage plan.
- **stage** (Required) - API stage name of the associated API stage in a usage plan.

### » Quota Settings Arguments

- **limit** (Optional) - The maximum number of requests that can be made in a given time period.
- **offset** (Optional) - The number of requests subtracted from the given limit in the initial time period.

- **period** (Optional) - The time period in which the limit applies. Valid values are "DAY", "WEEK" or "MONTH".

#### » Throttling Settings Arguments

- **burst\_limit** (Optional) - The API request burst limit, the maximum rate limit over a time ranging from one to a few seconds, depending upon whether the underlying token bucket is at its full capacity.
- **rate\_limit** (Optional) - The API request steady-state rate limit.

#### » Attributes Reference

The following attributes are exported:

- **id** - The ID of the API resource
- **name** - The name of the usage plan.
- **description** - The description of a usage plan.
- **api\_stages** - The associated API stages of the usage plan.
- **quota\_settings** - The quota of the usage plan.
- **throttle\_settings** - The throttling limits of the usage plan.
- **product\_code** - The AWS Marketplace product identifier to associate with the usage plan as a SaaS product on AWS Marketplace.

#### » Import

AWS API Gateway Usage Plan can be imported using the **id**, e.g.

```
$ terraform import aws_api_gateway_usage_plan.myusageplan <usage_plan_id>
```

#### » `aws_api_gateway_usage_plan_key`

Provides an API Gateway Usage Plan Key.

#### » Example Usage

```
resource "aws_api_gateway_rest_api" "test" {
  name = "MyDemoAPI"
}

...

resource "aws_api_gateway_usage_plan" "myusageplan" {
```

```

    name = "my_usage_plan"
  }

resource "aws_api_gateway_api_key" "mykey" {
  name = "my_key"

  stage_key {
    rest_api_id = "${aws_api_gateway_rest_api.test.id}"
    stage_name  = "${aws_api_gateway_deployment.foo.stage_name}"
  }
}

resource "aws_api_gateway_usage_plan_key" "main" {
  key_id      = "${aws_api_gateway_api_key.mykey.id}"
  key_type    = "API_KEY"
  usage_plan_id = "${aws_api_gateway_usage_plan.myusageplan.id}"
}

```

## » Argument Reference

The following arguments are supported:

- **key\_id** - (Required) The identifier of the API key resource.
- **key\_type** - (Required) The type of the API key resource. Currently, the valid key type is `API_KEY`.
- **usage\_plan\_id** - (Required) The Id of the usage plan resource representing to associate the key to.

## » Attributes Reference

The following attributes are exported:

- **id** - The Id of a usage plan key.
- **key\_id** - The identifier of the API gateway key resource.
- **key\_type** - The type of a usage plan key. Currently, the valid key type is `API_KEY`.
- **usage\_plan\_id** - The ID of the API resource
- **name** - The name of a usage plan key.
- **value** - The value of a usage plan key.

## » `aws_api_gateway_vpc_link`

Provides an API Gateway VPC Link.

## » Example Usage

```
resource "aws_lb" "example" {
  name = "example"
  internal = true
  load_balancer_type = "network"

  subnet_mapping {
    subnet_id = "12345"
  }
}

resource "aws_api_gateway_vpc_link" "example" {
  name = "example"
  description = "example description"
  target_arns = ["${aws_lb.example.arn}"]
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name used to label and identify the VPC link.
- **description** - (Optional) The description of the VPC link.
- **target\_arns** - (Required, ForceNew) The list of network load balancer arns in the VPC targeted by the VPC link. Currently AWS only supports 1 target.

## » Attributes Reference

The following attributes are exported:

- **id** - The identifier of the VpcLink.

## » Import

API Gateway VPC Link can be imported using the **id**, e.g.

```
$ terraform import aws_api_gateway_vpc_link.example <vpc_link_id>
```

## » `aws__appautoscaling__policy`

Provides an Application AutoScaling Policy resource.

## » Example Usage

### » DynamoDB Table Autoscaling

```
resource "aws_appautoscaling_target" "dynamodb_table_read_target" {
  max_capacity      = 100
  min_capacity      = 5
  resource_id       = "table/tableName"
  role_arn          = "${data.aws_iam_role.DynamoDBAutoscaleRole.arn}"
  scalable_dimension = "dynamodb:table:ReadCapacityUnits"
  service_namespace = "dynamodb"
}

resource "aws_appautoscaling_policy" "dynamodb_table_read_policy" {
  name                = "DynamoDBReadCapacityUtilization:${aws_appautoscaling_target.dynamodb_table_read_target.resource_id}"
  policy_type         = "TargetTrackingScaling"
  resource_id         = "${aws_appautoscaling_target.dynamodb_table_read_target.resource_id}"
  scalable_dimension  = "${aws_appautoscaling_target.dynamodb_table_read_target.scalable_dimension}"
  service_namespace   = "${aws_appautoscaling_target.dynamodb_table_read_target.service_namespace}"

  target_tracking_scaling_policy_configuration {
    predefined_metric_specification {
      predefined_metric_type = "DynamoDBReadCapacityUtilization"
    }

    target_value = 70
  }
}
```

### » ECS Service Autoscaling

```
resource "aws_appautoscaling_target" "ecs_target" {
  max_capacity      = 4
  min_capacity      = 1
  resource_id       = "service/clusterName/serviceName"
  role_arn          = "${var.ecs_iam_role}"
  scalable_dimension = "ecs:service:DesiredCount"
  service_namespace = "ecs"
}

resource "aws_appautoscaling_policy" "ecs_policy" {
  name                = "scale-down"
  policy_type         = "StepScaling"
  resource_id         = "service/clusterName/serviceName"
  scalable_dimension  = "ecs:service:DesiredCount"
}
```



```

service_namespace      = "ecs"

step_scaling_policy_configuration {
  adjustment_type      = "ChangeInCapacity"
  cooldown              = 60
  metric_aggregation_type = "Maximum"

  step_adjustment {
    metric_interval_upper_bound = 0
    scaling_adjustment          = -1
  }
}

depends_on = ["aws_appautoscaling_target.ecs_target"]
}

```

#### » Preserve desired count when updating an autoscaled ECS Service

```

resource "aws_ecs_service" "ecs_service" {
  name = "serviceName"
  cluster = "clusterName"
  task_definition = "taskDefinitionFamily:1"
  desired_count = 2

  lifecycle {
    ignore_changes = ["desired_count"]
  }
}

```

#### » Aurora Read Replica Autoscaling

```

resource "aws_appautoscaling_target" "replicas" {
  service_namespace = "rds"
  scalable_dimension = "rds:cluster:ReadReplicaCount"
  resource_id = "cluster:${aws_rds_cluster.example.id}"
  min_capacity = 1
  max_capacity = 15
}

resource "aws_appautoscaling_policy" "replicas" {
  name = "cpu-auto-scaling"
  service_namespace = "${aws_appautoscaling_target.replicas.service_namespace}"
  scalable_dimension = "${aws_appautoscaling_target.replicas.scalable_dimension}"
  resource_id = "${aws_appautoscaling_target.replicas.resource_id}"
}

```

```

policy_type          = "TargetTrackingScaling"

target_tracking_scaling_policy_configuration {
  predefined_metric_specification {
    predefined_metric_type = "RDSReaderAverageCPUUtilization"
  }
  target_value = 75
  scale_in_cooldown = 300
  scale_out_cooldown = 300
}
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the policy.
- **policy\_type** - (Optional) For DynamoDB, only `TargetTrackingScaling` is supported. For Amazon ECS, Spot Fleet, and Amazon RDS, both `StepScaling` and `TargetTrackingScaling` are supported. For any other service, only `StepScaling` is supported. Defaults to `StepScaling`.
- **resource\_id** - (Required) The resource type and unique identifier string for the resource associated with the scaling policy. Documentation can be found in the `ResourceId` parameter at: [AWS Application Auto Scaling API Reference](#)
- **scalable\_dimension** - (Required) The scalable dimension of the scalable target. Documentation can be found in the `ScalableDimension` parameter at: [AWS Application Auto Scaling API Reference](#)
- **service\_namespace** - (Required) The AWS service namespace of the scalable target. Documentation can be found in the `ServiceNamespace` parameter at: [AWS Application Auto Scaling API Reference](#)
- **step\_scaling\_policy\_configuration** - (Optional) Step scaling policy configuration, requires `policy_type = "StepScaling"` (default). See supported fields below.
- **target\_tracking\_scaling\_policy\_configuration** - (Optional) A target tracking policy, requires `policy_type = "TargetTrackingScaling"`. See supported fields below.

## » Nested fields

### » `step_scaling_policy_configuration`

- **adjustment\_type** - (Required) Specifies whether the adjustment is an absolute number or a percentage of the current capacity. Valid values are

ChangeInCapacity, ExactCapacity, and PercentChangeInCapacity.

- **cooldown** - (Required) The amount of time, in seconds, after a scaling activity completes and before the next scaling activity can start.
- **metric\_aggregation\_type** - (Optional) The aggregation type for the policy's metrics. Valid values are "Minimum", "Maximum", and "Average". Without a value, AWS will treat the aggregation type as "Average".
- **min\_adjustment\_magnitude** - (Optional) The minimum number to adjust your scalable dimension as a result of a scaling activity. If the adjustment type is PercentChangeInCapacity, the scaling policy changes the scalable dimension of the scalable target by this amount.
- **step\_adjustment** - (Optional) A set of adjustments that manage scaling. These have the following structure:

```
step_adjustment {  
    metric_interval_lower_bound = 1.0  
    metric_interval_upper_bound = 2.0  
    scaling_adjustment = -1  
}  
  
step_adjustment {  
    metric_interval_lower_bound = 2.0  
    metric_interval_upper_bound = 3.0  
    scaling_adjustment = 1  
}
```

- **metric\_interval\_lower\_bound** - (Optional) The lower bound for the difference between the alarm threshold and the CloudWatch metric. Without a value, AWS will treat this bound as infinity.
- **metric\_interval\_upper\_bound** - (Optional) The upper bound for the difference between the alarm threshold and the CloudWatch metric. Without a value, AWS will treat this bound as infinity. The upper bound must be greater than the lower bound.
- **scaling\_adjustment** - (Required) The number of members by which to scale, when the adjustment bounds are breached. A positive value scales up. A negative value scales down.

#### » **target\_tracking\_scaling\_policy\_configuration**

- **target\_value** - (Required) The target value for the metric.
- **disable\_scale\_in** - (Optional) Indicates whether scale in by the target tracking policy is disabled. If the value is true, scale in is disabled and the target tracking policy won't remove capacity from the scalable resource. Otherwise, scale in is enabled and the target tracking policy can remove capacity from the scalable resource. The default value is **false**.
- **scale\_in\_cooldown** - (Optional) The amount of time, in seconds, after a scale in activity completes before another scale in activity can start.

- **scale\_out\_cooldown** - (Optional) The amount of time, in seconds, after a scale out activity completes before another scale out activity can start.
- **customized\_metric\_specification** - (Optional) Reserved for future use. See supported fields below.
- **predefined\_metric\_specification** - (Optional) A predefined metric. See supported fields below.

#### » **customized\_metric\_specification**

- **dimensions** - (Optional) The dimensions of the metric.
- **metric\_name** - (Required) The name of the metric.
- **namespace** - (Required) The namespace of the metric.
- **statistic** - (Required) The statistic of the metric.
- **unit** - (Optional) The unit of the metric.

#### » **predefined\_metric\_specification**

- **predefined\_metric\_type** - (Required) The metric type.
- **resource\_label** - (Optional) Reserved for future use.

#### » **Attribute Reference**

- **adjustment\_type** - The scaling policy's adjustment type.
- **arn** - The ARN assigned by AWS to the scaling policy.
- **name** - The scaling policy's name.
- **policy\_type** - The scaling policy's type.

### » **aws\_\_appautoscaling\_\_scheduled\_\_action**

Provides an Application AutoScaling ScheduledAction resource.

#### » **Example Usage**

##### » **DynamoDB Table Autoscaling**

```
resource "aws_appautoscaling_target" "dynamodb" {
  max_capacity      = 100
  min_capacity      = 5
  resource_id       = "table/tableName"
  role_arn          = "${data.aws_iam_role.DynamoDBAutoscaleRole.arn}"
  scalable_dimension = "dynamodb:table:ReadCapacityUnits"
```

```

    service_namespace = "dynamodb"
}

resource "aws_appautoscaling_target" "dynamodb" {
  name = "dynamodb"
  service_namespace = "${aws_appautoscaling_target.dynamodb.service_namespace}"
  resource_id = "${aws_appautoscaling_target.dynamodb.resource_id}"
  scalable_dimension = "${aws_appautoscaling_target.dynamodb.scalable_dimension}"
  schedule = "at(2006-01-02T15:04:05)"

  scalable_target_action {
    min_capacity = 1
    max_capacity = 200
  }
}

```

## » ECS Service Autoscaling

```

resource "aws_appautoscaling_target" "ecs" {
  max_capacity      = 4
  min_capacity      = 1
  resource_id       = "service/clusterName/serviceName"
  role_arn          = "${var.ecs_iam_role}"
  scalable_dimension = "ecs:service:DesiredCount"
  service_namespace = "ecs"
}

resource "aws_appautoscaling_target" "ecs" {
  name = "ecs"
  service_namespace = "${aws_appautoscaling_target.ecs.service_namespace}"
  resource_id = "${aws_appautoscaling_target.ecs.resource_id}"
  scalable_dimension = "${aws_appautoscaling_target.ecs.scalable_dimension}"
  schedule = "at(2006-01-02T15:04:05)"

  scalable_target_action {
    min_capacity = 1
    max_capacity = 10
  }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the scheduled action.

- **service\_namespace** - (Required) The namespace of the AWS service. Documentation can be found in the parameter at: AWS Application Auto Scaling API Reference Example: ecs
- **resource\_id** - (Required) The identifier of the resource associated with the scheduled action. Documentation can be found in the parameter at: AWS Application Auto Scaling API Reference
- **scalable\_dimension** - (Optional) The scalable dimension. Documentation can be found in the parameter at: AWS Application Auto Scaling API Reference Example: ecs:service:DesiredCount
- **scalable\_target\_action** - (Optional) The new minimum and maximum capacity. You can set both values or just one. See below
- **schedule** - (Optional) The schedule for this action. The following formats are supported: At expressions - at(yyyy-mm-ddThh:mm:ss), Rate expressions - rate(valueunit), Cron expressions - cron(fields). In UTC. Documentation can be found in the parameter at: AWS Application Auto Scaling API Reference
- **start\_time** - (Optional) The date and time for the scheduled action to start. Specify the following format: 2006-01-02T15:04:05Z
- **end\_time** - (Optional) The date and time for the scheduled action to end. Specify the following format: 2006-01-02T15:04:05Z

#### » Scalable Target Action Arguments

- **max\_capacity** - (Optional) The maximum capacity.
- **min\_capacity** - (Optional) The minimum capacity.

#### » Attributes Reference

The following attributes are exported:

- **arn** - The Amazon Resource Name (ARN) of the scheduled action.

#### » **aws\_appautoscaling\_target**

Provides an Application AutoScaling ScalableTarget resource. To manage policies which get attached to the target, see the **aws\_appautoscaling\_policy** resource.

## » Example Usage

### » DynamoDB Table Autoscaling

```
resource "aws_appautoscaling_target" "dynamodb_table_read_target" {
  max_capacity      = 100
  min_capacity      = 5
  resource_id       = "table/${aws_dynamodb_table.example.name}"
  role_arn          = "${data.aws_iam_role.DynamoDBAutoscaleRole.arn}"
  scalable_dimension = "dynamodb:table:ReadCapacityUnits"
  service_namespace = "dynamodb"
}
```

### » DynamoDB Index Autoscaling

```
resource "aws_appautoscaling_target" "dynamodb_index_read_target" {
  max_capacity      = 100
  min_capacity      = 5
  resource_id       = "table/${aws_dynamodb_table.example.name}/index/${var.index_name}"
  role_arn          = "${data.aws_iam_role.DynamoDBAutoscaleRole.arn}"
  scalable_dimension = "dynamodb:index:ReadCapacityUnits"
  service_namespace = "dynamodb"
}
```

### » ECS Service Autoscaling

```
resource "aws_appautoscaling_target" "ecs_target" {
  max_capacity      = 4
  min_capacity      = 1
  resource_id       = "service/${aws_ecs_cluster.example.name}/${aws_ecs_service.example.name}"
  role_arn          = "${var.ecs_iam_role}"
  scalable_dimension = "ecs:service:DesiredCount"
  service_namespace = "ecs"
}
```

### » Aurora Read Replica Autoscaling

```
resource "aws_appautoscaling_target" "replicas" {
  service_namespace = "rds"
  scalable_dimension = "rds:cluster:ReadReplicaCount"
  resource_id       = "cluster:${aws_rds_cluster.example.id}"
  min_capacity      = 1
  max_capacity      = 15
}
```

}

## » Argument Reference

The following arguments are supported:

- **max\_capacity** - (Required) The max capacity of the scalable target.
- **min\_capacity** - (Required) The min capacity of the scalable target.
- **resource\_id** - (Required) The resource type and unique identifier string for the resource associated with the scaling policy. Documentation can be found in the **ResourceId** parameter at: [AWS Application Auto Scaling API Reference](#)
- **role\_arn** - (Optional) The ARN of the IAM role that allows Application AutoScaling to modify your scalable target on your behalf.
- **scalable\_dimension** - (Required) The scalable dimension of the scalable target. Documentation can be found in the **ScalableDimension** parameter at: [AWS Application Auto Scaling API Reference](#)
- **service\_namespace** - (Required) The AWS service namespace of the scalable target. Documentation can be found in the **ServiceNamespace** parameter at: [AWS Application Auto Scaling API Reference](#)

## » aws\_\_appsync\_\_datasource

Provides an AppSync DataSource.

## » Example Usage

```
resource "aws_dynamodb_table" "example" {
  name = "example"
  read_capacity = 1
  write_capacity = 1
  hash_key = "UserId"
  attribute {
    name = "UserId"
    type = "S"
  }
}

resource "aws_iam_role" "example" {
  name = "example"

  assume_role_policy = <<EOF
{
```



```

    "Version": "2012-10-17",
    "Statement": [
      {
        "Action": "sts:AssumeRole",
        "Principal": {
          "Service": "appsync.amazonaws.com"
        },
        "Effect": "Allow"
      }
    ]
  }
}
EOF
}

resource "aws_iam_role_policy" "example" {
  name = "example"
  role = "${aws_iam_role.example.id}"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "dynamodb:*"
      ],
      "Effect": "Allow",
      "Resource": [
        "${aws_dynamodb_table.example.arn}"
      ]
    }
  ]
}
}
EOF
}

resource "aws_appsync_graphql_api" "example" {
  authentication_type = "API_KEY"
  name = "tf_appsync_example"
}

resource "aws_appsync_datasource" "example" {
  api_id = "${aws_appsync_graphql_api.example.id}"
  name = "tf_appsync_example"
  type = "AMAZON_DYNAMODB"
  dynamodb_config {

```

```

    region = "us-west-2"
    table_name = "${aws_dynamodb_table.example.name}"
  }
  service_role_arn = "${aws_iam_role.example.arn}"
}

```

## » Argument Reference

The following arguments are supported:

- **api\_id** - (Required) The API ID for the GraphQL API for the DataSource.
- **name** - (Required) A user-supplied name for the DataSource.
- **type** - (Required) The type of the DataSource. Valid values: **AWS\_LAMBDA**, **AMAZON\_DYNAMODB** and **AMAZON\_ELASTICSEARCH**
- **description** - (Optional) A description of the DataSource.
- **service\_role\_arn** - (Optional) The IAM service role ARN for the data source.
- **dynamodb\_config** - (Optional) DynamoDB settings. See below
- **elasticsearch\_config** - (Optional) Amazon Elasticsearch settings. See below
- **lambda\_config** - (Optional) AWS Lambda settings. See below

### » dynamodb\_config

The following arguments are supported:

- **region** - (Required) The AWS region.
- **table\_name** - (Required) The table name.
- **use\_caller\_credentials** - (Optional) Set to TRUE to use Amazon Cognito credentials with this data source.

### » elasticsearch\_config

The following arguments are supported:

- **region** - (Required) The AWS region.
- **endpoint** - (Required) The endpoint.

### » lambda\_config

The following arguments are supported:

- **function\_arn** - (Required) The ARN for the Lambda function.

## » Attributes Reference

The following attributes are exported:

- **arn** - The ARN

## » **aws\_\_appsync\_graphql\_api**

Provides an AppSync GraphQL API.

## » Example Usage

```
resource "aws_appsync_graphql_api" "example" {
  authentication_type = "API_KEY"
  name = "example"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A user-supplied name for the GraphqlApi.
- **authentication\_type** - (Required) The authentication type. Valid values: `API_KEY`, `AWS_IAM` and `AMAZON_COGNITO_USER_POOLS`
- **user\_pool\_config** - (Optional) The Amazon Cognito User Pool configuration. See below

## » **user\_\_pool\_config**

The following arguments are supported:

- **aws\_region** - (Required) The AWS region in which the user pool was created.
- **default\_action** - (Required) The action that you want your GraphQL API to take when a request that uses Amazon Cognito User Pool authentication doesn't match the Amazon Cognito User Pool configuration. Valid: `ALLOW` and `DENY`
- **user\_pool\_id** - (Required) The user pool ID.
- **app\_id\_client\_regex** - (Optional) A regular expression for validating the incoming Amazon Cognito User Pool app client ID.

## » Attributes Reference

The following attributes are exported:

- `id` - API ID
- `arn` - The ARN

## » Import

AppSync GraphQL API can be imported using the GraphQL API ID, e.g.

```
$ terraform import aws_appsync_graphql_api.example 0123456789
```

## » `aws__athena__database`

Provides an Athena database.

## » Example Usage

```
resource "aws_s3_bucket" "hoge" {  
  bucket = "hoge"  
}  
  
resource "aws_athena_database" "hoge" {  
  name = "database_name"  
  bucket = "${aws_s3_bucket.hoge.bucket}"  
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the database to create.
- `bucket` - (Required) Name of s3 bucket to save the results of the query execution.
- `force_destroy` - (Optional, Default: `false`) A boolean that indicates all tables should be deleted from the database so that the database can be destroyed without error. The tables are *not* recoverable.

**NOTE:** When Athena queries are executed, result files may be created in the specified bucket. Consider using `force_destroy` on the bucket too in order to avoid any problems when destroying the bucket.

## » Attributes Reference

The following attributes are exported:

- `id` - The database name

## » `aws_athena_named_query`

Provides an Athena Named Query resource.

## » Example Usage

```
resource "aws_s3_bucket" "hoge" {
  bucket = "tf-test"
}

resource "aws_athena_database" "hoge" {
  name = "users"
  bucket = "${aws_s3_bucket.hoge.bucket}"
}

resource "aws_athena_named_query" "foo" {
  name = "bar"
  database = "${aws_athena_database.hoge.name}"
  query = "SELECT * FROM ${aws_athena_database.hoge.name} limit 10;"
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) The plain language name for the query. Maximum length of 128.
- `database` - (Required) The database to which the query belongs.
- `query` - (Required) The text of the query itself. In other words, all query statements. Maximum length of 262144.
- `description` - (Optional) A brief explanation of the query. Maximum length of 1024.

## » Attributes Reference

The following attributes are exported:

- `id` - The unique ID of the query.

## » Import

Athena Named Query can be imported using the query ID, e.g.

```
$ terraform import aws_athena_named_query.example 0123456789
```

## » `aws_batch_compute_environment`

Creates a AWS Batch compute environment. Compute environments contain the Amazon ECS container instances that are used to run containerized batch jobs.

For information about AWS Batch, see [What is AWS Batch?](#) . For information about compute environment, see [Compute Environments](#) .

**Note:** To prevent a race condition during environment deletion, make sure to set `depends_on` to the related `aws_iam_role_policy_attachment`; otherwise, the policy may be destroyed too soon and the compute environment will then get stuck in the `DELETING` state, see [Troubleshooting AWS Batch](#) .

## » Example Usage

```
resource "aws_iam_role" "ecs_instance_role" {
  name = "ecs_instance_role"
  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Effect": "Allow",
      "Principal": {
        "Service": "ec2.amazonaws.com"
      }
    }
  ]
}
EOF
}

resource "aws_iam_role_policy_attachment" "ecs_instance_role" {
  role      = "${aws_iam_role.ecs_instance_role.name}"
```

```

    policy_arn = "arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role"
}

resource "aws_iam_instance_profile" "ecs_instance_role" {
    name = "ecs_instance_role"
    role = "${aws_iam_role.ecs_instance_role.name}"
}

resource "aws_iam_role" "aws_batch_service_role" {
    name = "aws_batch_service_role"
    assume_role_policy = <<EOF
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "sts:AssumeRole",
            "Effect": "Allow",
            "Principal": {
                "Service": "batch.amazonaws.com"
            }
        }
    ]
}
}
EOF
}

resource "aws_iam_role_policy_attachment" "aws_batch_service_role" {
    role = "${aws_iam_role.aws_batch_service_role.name}"
    policy_arn = "arn:aws:iam::aws:policy/service-role/AWSBatchServiceRole"
}

resource "aws_security_group" "sample" {
    name = "aws_batch_compute_environment_security_group"
}

resource "aws_vpc" "sample" {
    cidr_block = "10.1.0.0/16"
}

resource "aws_subnet" "sample" {
    vpc_id = "${aws_vpc.sample.id}"
    cidr_block = "10.1.1.0/24"
}

resource "aws_batch_compute_environment" "sample" {
    compute_environment_name = "sample"
}

```

```

compute_resources {
  instance_role = "${aws_iam_instance_profile.ecs_instance_role.arn}"
  instance_type = [
    "c4.large",
  ]
  max_vcpus = 16
  min_vcpus = 0
  security_group_ids = [
    "${aws_security_group.sample.id}"
  ]
  subnets = [
    "${aws_subnet.sample.id}"
  ]
  type = "EC2"
}
service_role = "${aws_iam_role.aws_batch_service_role.arn}"
type = "MANAGED"
depends_on = ["aws_iam_role_policy_attachment.aws_batch_service_role"]
}

```

## » Argument Reference

- **compute\_environment\_name** - (Required) The name for your compute environment. Up to 128 letters (uppercase and lowercase), numbers, and underscores are allowed.
- **compute\_resources** - (Optional) Details of the compute resources managed by the compute environment. This parameter is required for managed compute environments. See details below.
- **service\_role** - (Required) The full Amazon Resource Name (ARN) of the IAM role that allows AWS Batch to make calls to other AWS services on your behalf.
- **state** - (Optional) The state of the compute environment. If the state is **ENABLED**, then the compute environment accepts jobs from a queue and can scale out automatically based on queues. Valid items are **ENABLED** or **DISABLED**. Defaults to **ENABLED**.
- **type** - (Required) The type of the compute environment. Valid items are **MANAGED** or **UNMANAGED**.

**compute\_\_resources** is a child block with a single argument:

- **bid\_percentage** - (Optional) The minimum percentage that a Spot Instance price must be when compared with the On-Demand price for that instance type before instances are launched. For example, if your bid percentage is 20%, then the Spot price must be below 20% of the current On-Demand price for that EC2 instance. This parameter is required for SPOT compute environments.



- **desired\_vcpus** - (Optional) The desired number of EC2 vCPUS in the compute environment.
- **ec2\_key\_pair** - (Optional) The EC2 key pair that is used for instances launched in the compute environment.
- **image\_id** - (Optional) The Amazon Machine Image (AMI) ID used for instances launched in the compute environment.
- **instance\_role** - (Required) The Amazon ECS instance role applied to Amazon EC2 instances in a compute environment.
- **instance\_type** - (Required) A list of instance types that may be launched.
- **max\_vcpus** - (Required) The maximum number of EC2 vCPUs that an environment can reach.
- **min\_vcpus** - (Required) The minimum number of EC2 vCPUs that an environment should maintain.
- **security\_group\_ids** - (Required) A list of EC2 security group that are associated with instances launched in the compute environment.
- **spot\_iam\_fleet\_role** - (Optional) The Amazon Resource Name (ARN) of the Amazon EC2 Spot Fleet IAM role applied to a SPOT compute environment. This parameter is required for SPOT compute environments.
- **subnets** - (Required) A list of VPC subnets into which the compute resources are launched.
- **tags** - (Optional) Key-value pair tags to be applied to resources that are launched in the compute environment.
- **type** - (Required) The type of compute environment. Valid items are EC2 or SPOT.

## » Attributes Reference

- **arn** - The Amazon Resource Name (ARN) of the compute environment.
- **ecs\_cluster\_arn** - The Amazon Resource Name (ARN) of the underlying Amazon ECS cluster used by the compute environment.
- **status** - The current status of the compute environment (for example, CREATING or VALID).
- **status\_reason** - A short, human-readable string to provide additional details about the current status of the compute environment.

## » `aws_batch_job_definition`

Provides a Batch Job Definition resource.

## » Example Usage

```
resource "aws_batch_job_definition" "test" {
```

```

    name = "tf_test_batch_job_definition"
    type = "container"
    container_properties = <<CONTAINER_PROPERTIES
  {
    "command": ["ls", "-la"],
    "image": "busybox",
    "memory": 1024,
    "vcpus": 1,
    "volumes": [
      {
        "host": {
          "sourcePath": "/tmp"
        },
        "name": "tmp"
      }
    ],
    "environment": [
      {"name": "VARNAME", "value": "VARVAL"}
    ],
    "mountPoints": [
      {
        "sourceVolume": "tmp",
        "containerPath": "/tmp",
        "readOnly": false
      }
    ],
    "ulimits": [
      {
        "hardLimit": 1024,
        "name": "nofile",
        "softLimit": 1024
      }
    ]
  }
CONTAINER_PROPERTIES
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) Specifies the name of the job definition.
- **container\_properties** - (Optional) A valid container properties provided as a single valid JSON document. This parameter is required if the **type** parameter is **container**.

- **parameters** - (Optional) Specifies the parameter substitution placeholders to set in the job definition.
- **retry\_strategy** - (Optional) Specifies the retry strategy to use for failed jobs that are submitted with this job definition. Maximum number of **retry\_strategy** is 1. Defined below.
- **timeout** - (Optional) Specifies the timeout for jobs so that if a job runs longer, AWS Batch terminates the job. Maximum number of **timeout** is 1. Defined below.
- **type** - (Required) The type of job definition. Must be **container**

#### » **retry\_strategy**

**retry\_strategy** supports the following:

- **attempts** - (Optional) The number of times to move a job to the **RUNNABLE** status. You may specify between 1 and 10 attempts.

#### » **timeout**

**timeout** supports the following:

- **attempt\_duration\_seconds** - (Optional) The time duration in seconds after which AWS Batch terminates your jobs if they have not finished. The minimum value for the timeout is 60 seconds.

### » **Attribute Reference**

The following attributes are exported:

- **arn** - The Amazon Resource Name of the job definition.
- **revision** - The revision of the job definition.

#### » **aws\_batch\_job\_queue**

Provides a Batch Job Queue resource.

#### » **Example Usage**

```
resource "aws_batch_job_queue" "test_queue" {
  name = "tf-test-batch-job-queue"
  state = "ENABLED"
  priority = 1
}
```

```
compute_environments = ["${aws_batch_compute_environment.test_environment_1.arn}", "${aws.
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) Specifies the name of the job queue.
- **compute\_environments** - (Required) Specifies the set of compute environments mapped to a job queue and their order. The position of the compute environments in the list will dictate the order. You can associate up to 3 compute environments with a job queue.
- **priority** - (Required) The priority of the job queue. Job queues with a higher priority are evaluated first when associated with the same compute environment.
- **state** - (Required) The state of the job queue. Must be one of: `ENABLED` or `DISABLED`

## » Attribute Reference

The following attributes are exported:

- **arn** - The Amazon Resource Name of the job queue.

## » `aws_cloud9_environment_ec2`

Provides a Cloud9 EC2 Development Environment.

## » Example Usage

```
resource "aws_cloud9_environment_ec2" "example" {
  instance_type = "t2.micro"
  name = "example-env"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the environment.
- **instance\_type** - (Required) The type of instance to connect to the environment, e.g. `t2.micro`.

- `automatic_stop_time_minutes` - (Optional) The number of minutes until the running instance is shut down after the environment has last been used.
- `description` - (Optional) The description of the environment.
- `owner_arn` - (Optional) The ARN of the environment owner. This can be ARN of any AWS IAM principal. Defaults to the environment's creator.
- `subnet_id` - (Optional) The ID of the subnet in Amazon VPC that AWS Cloud9 will use to communicate with the Amazon EC2 instance.

## » Attributes Reference

In addition the the arguments listed above the following attributes are exported:

- `id` - The ID of the environment.
- `arn` - The ARN of the environment.
- `type` - The type of the environment (e.g. `ssh` or `ec2`)

## » `aws__budgets__budget`

Provides a budgets budget resource. Budgets use the cost visualisation provided by Cost Explorer to show you the status of your budgets, to provide forecasts of your estimated costs, and to track your AWS usage, including your free tier usage.

## » Example Usage

```
resource "aws_budgets_budget" "ec2" {
  name           = "budget-ec2-monthly"
  budget_type    = "COST"
  limit_amount   = "1200"
  limit_unit     = "USD"
  time_period_end = "2087-06-15_00:00"
  time_period_start = "2017-07-01_00:00"
  time_unit      = "MONTHLY"

  cost_filters {
    service = "ec2"
  }
}
```

Create a budget for *\$100*.

```
resource "aws_budgets_budget" "cost" {
  ...
  budget_type = "COST"
}
```

```

    limit_amount = "100"
    limit_unit   = "USD"
}

```

Create a budget for s3 with a limit of 3 GB of storage.

```

resource "aws_budgets_budget" "s3" {
  ...
  budget_type = "USAGE"
  limit_amount = "3"
  limit_unit   = "GB"
}

```

## » Argument Reference

For more detailed documentation about each argument, refer to the AWS official documentation.

The following arguments are supported:

- **account\_id** - (Optional) The ID of the target account for budget. Will use current user's account\_id by default if omitted.
- **name** - (Optional) The name of a budget. Unique within accounts.
- **name\_prefix** - (Optional) The prefix of the name of a budget. Unique within accounts.
- **budget\_type** - (Required) Whether this budget tracks monetary cost or usage.
- **cost\_filters** - (Optional) Map of CostFilters key/value pairs to apply to the budget.
- **cost\_types** - (Optional) Object containing CostTypes The types of cost included in a budget, such as tax and subscriptions..
- **limit\_amount** - (Required) The amount of cost or usage being measured for a budget.
- **limit\_unit** - (Required) The unit of measurement used for the budget forecast, actual spend, or budget threshold, such as dollars or GB. See Spend documentation.
- **time\_period\_end** - (Optional) The end of the time period covered by the budget. There are no restrictions on the end date. Format: 2017-01-01\_12:00.
- **time\_period\_start** - (Required) The start of the time period covered by the budget. The start date must come before the end date. Format: 2017-01-01\_12:00.
- **time\_unit** - (Required) The length of time until a budget resets the actual and forecasted spend. Valid values: MONTHLY, QUARTERLY, ANNUALLY.

## » Attributes Reference

The following attributes are exported:

- `id` - id of resource.

## » CostTypes

Valid keys for `cost_types` parameter.

- `include_credit` - A boolean value whether to include credits in the cost budget. Defaults to `true`
- `include_discount` - Specifies whether a budget includes discounts. Defaults to `true`
- `include_other_subscription` - A boolean value whether to include other subscription costs in the cost budget. Defaults to `true`
- `include_recurring` - A boolean value whether to include recurring costs in the cost budget. Defaults to `true`
- `include_refund` - A boolean value whether to include refunds in the cost budget. Defaults to `true`
- `include_subscription` - A boolean value whether to include subscriptions in the cost budget. Defaults to `true`
- `include_support` - A boolean value whether to include support costs in the cost budget. Defaults to `true`
- `include_tax` - A boolean value whether to include tax in the cost budget. Defaults to `true`
- `include_upfront` - A boolean value whether to include upfront costs in the cost budget. Defaults to `true`
- `use_amortized` - Specifies whether a budget uses the amortized rate. Defaults to `false`
- `use_blended` - A boolean value whether to use blended costs in the cost budget. Defaults to `false`

Refer to AWS CostTypes documentation for further detail.

## » CostFilters

Valid keys for `cost_filters` parameter vary depending on the `budget_type` value.

- `cost`
  - `AZ`
  - `LinkedAccount`
  - `Operation`
  - `PurchaseType`
  - `Service`

- TagKeyValue
- usage
  - AZ
  - LinkedAccount
  - Operation
  - PurchaseType
  - UsageType:<service name>
  - TagKeyValue

Refer to AWS CostFilter documentation for further detail.

## » Import

Budgets can be imported using AccountID:BudgetName, e.g.

```
$ terraform import aws_budgets_budget.myBudget 123456789012:myBudget
```

## » aws\_\_cloudformation\_\_stack

Provides a CloudFormation Stack resource.

## » Example Usage

```
resource "aws_cloudformation_stack" "network" {
  name = "networking-stack"

  parameters {
    VPCCidr = "10.0.0.0/16"
  }

  template_body = <<STACK
{
  "Parameters" : {
    "VPCCidr" : {
      "Type" : "String",
      "Default" : "10.0.0.0/16",
      "Description" : "Enter the CIDR block for the VPC. Default is 10.0.0.0/16."
    }
  },
  "Resources" : {
    "my-vpc": {
      "Type" : "AWS::EC2::VPC",
      "Properties" : {
```



```

        "CidrBlock" : { "Ref" : "VPCCidr" },
        "Tags" : [
            {"Key": "Name", "Value": "Primary_CF_VPC"}
        ]
    }
}
}
STACK
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) Stack name.
- **template\_body** - (Optional) Structure containing the template body (max size: 51,200 bytes).
- **template\_url** - (Optional) Location of a file containing the template body (max size: 460,800 bytes).
- **capabilities** - (Optional) A list of capabilities. Valid values: `CAPABILITY_IAM` or `CAPABILITY_NAMED_IAM`
- **disable\_rollback** - (Optional) Set to true to disable rollback of the stack if stack creation failed. Conflicts with **on\_failure**.
- **notification\_arns** - (Optional) A list of SNS topic ARNs to publish stack related events.
- **on\_failure** - (Optional) Action to be taken if stack creation fails. This must be one of: `DO_NOTHING`, `ROLLBACK`, or `DELETE`. Conflicts with **disable\_rollback**.
- **parameters** - (Optional) A list of Parameter structures that specify input parameters for the stack.
- **policy\_body** - (Optional) Structure containing the stack policy body. Conflicts w/ **policy\_url**.
- **policy\_url** - (Optional) Location of a file containing the stack policy. Conflicts w/ **policy\_body**.
- **tags** - (Optional) A list of tags to associate with this stack.
- **iam\_role\_arn** - (Optional) The ARN of an IAM role that AWS CloudFormation assumes to create the stack. If you don't specify a value, AWS CloudFormation uses the role that was previously associated with the stack. If no role is available, AWS CloudFormation uses a temporary session that is generated from your user credentials.
- **timeout\_in\_minutes** - (Optional) The amount of time that can pass before the stack status becomes `CREATE_FAILED`.

## » Attributes Reference

The following attributes are exported:

- `id` - A unique identifier of the stack.
- `outputs` - A map of outputs from the stack.

## » Import

Cloudformation Stacks can be imported using the `name`, e.g.

```
$ terraform import aws_cloudformation_stack.stack networking-stack
```

## » Timeouts

`aws_cloudformation_stack` provides the following Timeouts configuration options:

- `create` - (Default 30 minutes) Used for Creating Stacks
- `update` - (Default 30 minutes) Used for Stack modifications
- `delete` - (Default 30 minutes) Used for destroying stacks.

## » `aws__cloudfront__distribution`

Creates an Amazon CloudFront web distribution.

For information about CloudFront distributions, see the Amazon CloudFront Developer Guide. For specific information about creating CloudFront web distributions, see the POST Distribution page in the Amazon CloudFront API Reference.

**NOTE:** CloudFront distributions take about 15 minutes to a deployed state after creation or modification. During this time, deletes to resources will be blocked. If you need to delete a distribution that is enabled and you do not want to wait, you need to use the `retain_on_delete` flag.

## » Example Usage

The following example below creates a CloudFront distribution with an S3 origin.

```
resource "aws_s3_bucket" "b" {  
  bucket = "mybucket"  
  acl    = "private"
```

```

tags {
  Name = "My bucket"
}
}

resource "aws_cloudfront_distribution" "s3_distribution" {
  origin {
    domain_name = "${aws_s3_bucket.b.bucket_domain_name}"
    origin_id   = "myS3Origin"

    s3_origin_config {
      origin_access_identity = "origin-access-identity/cloudfront/ABCDEFG1234567"
    }
  }

  enabled          = true
  is_ipv6_enabled  = true
  comment          = "Some comment"
  default_root_object = "index.html"

  logging_config {
    include_cookies = false
    bucket          = "mylogs.s3.amazonaws.com"
    prefix          = "myprefix"
  }

  aliases = ["mysite.example.com", "yoursite.example.com"]

  default_cache_behavior {
    allowed_methods  = ["DELETE", "GET", "HEAD", "OPTIONS", "PATCH", "POST", "PUT"]
    cached_methods  = ["GET", "HEAD"]
    target_origin_id = "myS3Origin"

    forwarded_values {
      query_string = false

      cookies {
        forward = "none"
      }
    }
  }

  viewer_protocol_policy = "allow-all"
  min_ttl                = 0
  default_ttl            = 3600
  max_ttl                = 86400

```

```

}

# Cache behavior with precedence 0
ordered_cache_behavior {
    path_pattern      = "/content/immutable/*"
    allowed_methods   = ["GET", "HEAD", "OPTIONS"]
    cached_methods    = ["GET", "HEAD", "OPTIONS"]
    target_origin_id  = "myS3Origin"

    forwarded_values {
        query_string = false
        headers      = ["Origin"]
        cookies {
            forward = "none"
        }
    }
}

min_ttl      = 0
default_ttl  = 86400
max_ttl      = 31536000
compress     = true
viewer_protocol_policy = "redirect-to-https"
}

# Cache behavior with precedence 1
ordered_cache_behavior {
    path_pattern      = "/content/*"
    allowed_methods   = ["GET", "HEAD", "OPTIONS"]
    cached_methods    = ["GET", "HEAD"]
    target_origin_id  = "myS3Origin"

    forwarded_values {
        query_string = false
        cookies {
            forward = "none"
        }
    }
}

min_ttl      = 0
default_ttl  = 3600
max_ttl      = 86400
compress     = true
viewer_protocol_policy = "redirect-to-https"
}

price_class = "PriceClass_200"

```

```

restrictions {
  geo_restriction {
    restriction_type = "whitelist"
    locations       = ["US", "CA", "GB", "DE"]
  }
}

tags {
  Environment = "production"
}

viewer_certificate {
  cloudfront_default_certificate = true
}
}

```

## » Argument Reference

The CloudFront distribution argument layout is a complex structure composed of several sub-resources - these resources are laid out below.

### » Top-Level Arguments

- **aliases** (Optional) - Extra CNAMEs (alternate domain names), if any, for this distribution.
- **cache\_behavior** (Optional) - **Deprecated**, use **ordered\_cache\_behavior** instead.
- **ordered\_cache\_behavior** (Optional) - An ordered list of cache behaviors resource for this distribution. List from top to bottom
  - in order of precedence. The topmost cache behavior will have precedence 0.
- **comment** (Optional) - Any comments you want to include about the distribution.
- **custom\_error\_response** (Optional) - One or more custom error response elements (multiples allowed).
- **default\_cache\_behavior** (Required) - The default cache behavior for this distribution (maximum one).
- **default\_root\_object** (Optional) - The object that you want CloudFront to return (for example, index.html) when an end user requests the root URL.

- **enabled** (Required) - Whether the distribution is enabled to accept end user requests for content.
- **is\_ipv6\_enabled** (Optional) - Whether the IPv6 is enabled for the distribution.
- **http\_version** (Optional) - The maximum HTTP version to support on the distribution. Allowed values are **http1.1** and **http2**. The default is **http2**.
- **logging\_config** (Optional) - The logging configuration that controls how logs are written to your distribution (maximum one).
- **origin** (Required) - One or more origins for this distribution (multiples allowed).
- **price\_class** (Optional) - The price class for this distribution. One of **PriceClass\_All**, **PriceClass\_200**, **PriceClass\_100**
- **restrictions** (Required) - The restriction configuration for this distribution (maximum one).
- **tags** - (Optional) A mapping of tags to assign to the resource.
- **viewer\_certificate** (Required) - The SSL configuration for this distribution (maximum one).
- **web\_acl\_id** (Optional) - If you're using AWS WAF to filter CloudFront requests, the Id of the AWS WAF web ACL that is associated with the distribution.
- **retain\_on\_delete** (Optional) - Disables the distribution instead of deleting it when destroying the resource through Terraform. If this is set, the distribution needs to be deleted manually afterwards. Default: **false**.

#### » Cache Behavior Arguments

- **allowed\_methods** (Required) - Controls which HTTP methods CloudFront processes and forwards to your Amazon S3 bucket or your custom origin.
- **cached\_methods** (Required) - Controls whether CloudFront caches the response to requests using the specified HTTP methods.
- **compress** (Optional) - Whether you want CloudFront to automatically compress content for web requests that include **Accept-Encoding: gzip** in the request header (default: **false**).
- **default\_ttl** (Optional) - The default amount of time (in seconds) that an object is in a CloudFront cache before CloudFront forwards another request in the absence of an **Cache-Control max-age** or **Expires** header. Defaults to 1 day.

- **field\_level\_encryption\_id** (Optional) - Field level encryption configuration ID
- **forwarded\_values** (Required) - The forwarded values configuration that specifies how CloudFront handles query strings, cookies and headers (maximum one).
- **lambda\_function\_association** (Optional) - A config block that triggers a lambda function with specific actions. Defined below, maximum 4.
- **max\_ttl** (Optional) - The maximum amount of time (in seconds) that an object is in a CloudFront cache before CloudFront forwards another request to your origin to determine whether the object has been updated. Only effective in the presence of **Cache-Control max-age**, **Cache-Control s-maxage**, and **Expires** headers. Defaults to 365 days.
- **min\_ttl** (Optional) - The minimum amount of time that you want objects to stay in CloudFront caches before CloudFront queries your origin to see whether the object has been updated. Defaults to 0 seconds.
- **path\_pattern** (Required) - The pattern (for example, **images/\*.jpg**) that specifies which requests you want this cache behavior to apply to.
- **smooth\_streaming** (Optional) - Indicates whether you want to distribute media files in Microsoft Smooth Streaming format using the origin that is associated with this cache behavior.
- **target\_origin\_id** (Required) - The value of ID for the origin that you want CloudFront to route requests to when a request matches the path pattern either for a cache behavior or for the default cache behavior.
- **trusted\_signers** (Optional) - The AWS accounts, if any, that you want to allow to create signed URLs for private content.
- **viewer\_protocol\_policy** (Required) - Use this element to specify the protocol that users can use to access the files in the origin specified by TargetOriginId when a request matches the path pattern in PathPattern. One of **allow-all**, **https-only**, or **redirect-to-https**.

#### » Forwarded Values Arguments

- **cookies** (Required) - The forwarded values cookies that specifies how CloudFront handles cookies (maximum one).
- **headers** (Optional) - Specifies the Headers, if any, that you want CloudFront to vary upon for this cache behavior. Specify **\*** to include all headers.
- **query\_string** (Required) - Indicates whether you want CloudFront to forward query strings to the origin that is associated with this cache behavior.

- **query\_string\_cache\_keys** (Optional) - When specified, along with a value of **true** for **query\_string**, all query strings are forwarded, however only the query string keys listed in this argument are cached. When omitted with a value of **true** for **query\_string**, all query string keys are cached.

#### » **Lambda Function Association**

Lambda@Edge allows you to associate an AWS Lambda Function with a pre-defined event. You can associate a single function per event type. See What is Lambda@Edge for more information

- **event\_type** (Required) - The specific event to trigger this function. Valid values: **viewer-request**, **origin-request**, **viewer-response**, **origin-response**
- **lambda\_arn** (Required) - ARN of the Lambda function.

#### » **Cookies Arguments**

- **forward** (Required) - Specifies whether you want CloudFront to forward cookies to the origin that is associated with this cache behavior. You can specify **all**, **none** or **whitelist**. If **whitelist**, you must include the subsequent **whitelisted\_names**
- **whitelisted\_names** (Optional) - If you have specified **whitelist** to **forward**, the whitelisted cookies that you want CloudFront to forward to your origin.

#### » **Custom Error Response Arguments**

- **error\_caching\_min\_ttl** (Optional) - The minimum amount of time you want HTTP error codes to stay in CloudFront caches before CloudFront queries your origin to see whether the object has been updated.
- **error\_code** (Required) - The 4xx or 5xx HTTP status code that you want to customize.
- **response\_code** (Optional) - The HTTP status code that you want CloudFront to return with the custom error page to the viewer.
- **response\_page\_path** (Optional) - The path of the custom error page (for example, **/custom\_404.html**).

#### » **Default Cache Behavior Arguments**

The arguments for **default\_cache\_behavior** are the same as for **cache\_behavior**, except for the **path\_pattern** argument is not required.



#### » Logging Config Arguments

- **bucket** (Required) - The Amazon S3 bucket to store the access logs in, for example, `myawslogbucket.s3.amazonaws.com`.
- **include\_cookies** (Optional) - Specifies whether you want CloudFront to include cookies in access logs (default: `false`).
- **prefix** (Optional) - An optional string that you want CloudFront to prefix to the access log filenames for this distribution, for example, `myprefix/`.

#### » Origin Arguments

- **custom\_origin\_config** - The CloudFront custom origin configuration information. If an S3 origin is required, use **s3\_origin\_config** instead.
- **domain\_name** (Required) - The DNS domain name of either the S3 bucket, or web site of your custom origin.
- **custom\_header** (Optional) - One or more sub-resources with **name** and **value** parameters that specify header data that will be sent to the origin (multiples allowed).
- **origin\_id** (Required) - A unique identifier for the origin.
- **origin\_path** (Optional) - An optional element that causes CloudFront to request your content from a directory in your Amazon S3 bucket or your custom origin.
- **s3\_origin\_config** - The CloudFront S3 origin configuration information. If a custom origin is required, use **custom\_origin\_config** instead.

#### » Custom Origin Config Arguments

- **http\_port** (Required) - The HTTP port the custom origin listens on.
- **https\_port** (Required) - The HTTPS port the custom origin listens on.
- **origin\_protocol\_policy** (Required) - The origin protocol policy to apply to your origin. One of `http-only`, `https-only`, or `match-viewer`.
- **origin\_ssl\_protocols** (Required) - The SSL/TLS protocols that you want CloudFront to use when communicating with your origin over HTTPS. A list of one or more of `SSLv3`, `TLSv1`, `TLSv1.1`, and `TLSv1.2`.
- **origin\_keepalive\_timeout** - (Optional) The Custom KeepAlive timeout, in seconds. By default, AWS enforces a limit of 60. But you can request an increase.
- **origin\_read\_timeout** - (Optional) The Custom Read timeout, in seconds. By default, AWS enforces a limit of 60. But you can request an increase.

### » S3 Origin Config Arguments

- `origin_access_identity` (Optional) - The CloudFront origin access identity to associate with the origin.

### » Restrictions Arguments

The `restrictions` sub-resource takes another single sub-resource named `geo_restriction` (see the example for usage).

The arguments of `geo_restriction` are:

- `locations` (Optional) - The ISO 3166-1-alpha-2 codes for which you want CloudFront either to distribute your content (`whitelist`) or not distribute your content (`blacklist`).
- `restriction_type` (Required) - The method that you want to use to restrict distribution of your content by country: `none`, `whitelist`, or `blacklist`.

### » Viewer Certificate Arguments

- `acm_certificate_arn` - The ARN of the AWS Certificate Manager certificate that you wish to use with this distribution. Specify this, `cloudfront_default_certificate`, or `iam_certificate_id`. The ACM certificate must be in US-EAST-1.
- `cloudfront_default_certificate` - `true` if you want viewers to use HTTPS to request your objects and you're using the CloudFront domain name for your distribution. Specify this, `acm_certificate_arn`, or `iam_certificate_id`.
- `iam_certificate_id` - The IAM certificate identifier of the custom viewer certificate for this distribution if you are using a custom domain. Specify this, `acm_certificate_arn`, or `cloudfront_default_certificate`.
- `minimum_protocol_version` - The minimum version of the SSL protocol that you want CloudFront to use for HTTPS connections. One of `SSLv3`, `TLSv1`, `TLSv1_2016`, `TLSv1.1_2016` or `TLSv1.2_2018`. Default: `TLSv1`. **NOTE:** If you are using a custom certificate (specified with `acm_certificate_arn` or `iam_certificate_id`), and have specified `sni-only` in `ssl_support_method`, `TLSv1` or later must be specified. If you have specified `vip` in `ssl_support_method`, only `SSLv3` or `TLSv1` can be specified. If you have specified `cloudfront_default_certificate`, `TLSv1` must be specified.
- `ssl_support_method`: Specifies how you want CloudFront to serve HTTPS requests. One of `vip` or `sni-only`. Required if you specify

`acm_certificate_arn` or `iam_certificate_id`. **NOTE:** `vip` causes CloudFront to use a dedicated IP address and may incur extra charges.

## » Attribute Reference

The following attributes are exported:

- `id` - The identifier for the distribution. For example: `EDFDVBD632BHDS5`.
- `arn` - The ARN (Amazon Resource Name) for the distribution. For example: `arn:aws:cloudfront::123456789012:distribution/EDFDVBD632BHDS5`, where 123456789012 is your AWS account ID.
- `caller_reference` - Internal value used by CloudFront to allow future updates to the distribution configuration.
- `status` - The current status of the distribution. **Deployed** if the distribution's information is fully propagated throughout the Amazon CloudFront system.
- `active_trusted_signers` - The key pair IDs that CloudFront is aware of for each trusted signer, if the distribution is set up to serve private content with signed URLs.
- `domain_name` - The domain name corresponding to the distribution. For example: `d604721fxaaqy9.cloudfront.net`.
- `last_modified_time` - The date and time the distribution was last modified.
- `in_progress_validation_batches` - The number of invalidation batches currently in progress.
- `etag` - The current version of the distribution's information. For example: `E2QWRUHAPOMQZL`.
- `hosted_zone_id` - The CloudFront Route 53 zone ID that can be used to route an Alias Resource Record Set to. This attribute is simply an alias for the zone ID `Z2FDTNDATAQYW2`.

## » Import

Cloudfront Distributions can be imported using the `id`, e.g.

```
$ terraform import aws_cloudfront_distribution.distribution E74FTE3EXAMPLE
```

## » **aws\_\_cloudfront\_\_origin\_\_access\_\_identity**

Creates an Amazon CloudFront origin access identity.

For information about CloudFront distributions, see the Amazon CloudFront Developer Guide. For more information on generating origin access identities, see Using an Origin Access Identity to Restrict Access to Your Amazon S3 Content.

### » **Example Usage**

The following example below creates a CloudFront origin access identity.

```
resource "aws_cloudfront_origin_access_identity" "origin_access_identity" {  
  comment = "Some comment"  
}
```

### » **Argument Reference**

- **comment** (Optional) - An optional comment for the origin access identity.

### » **Attribute Reference**

The following attributes are exported:

- **id** - The identifier for the distribution. For example: EDFDVBD632BHDS5.
- **caller\_reference** - Internal value used by CloudFront to allow future updates to the origin access identity.
- **cloudfront\_access\_identity\_path** - A shortcut to the full path for the origin access identity to use in CloudFront, see below.
- **etag** - The current version of the origin access identity's information. For example: E2QWRUHAPOMQZL.
- **iam\_arn** - A pre-generated ARN for use in S3 bucket policies (see below). Example: `arn:aws:iam::cloudfront:user/CloudFront Origin Access Identity E2QWRUHAPOMQZL`.
- **s3\_canonical\_user\_id** - The Amazon S3 canonical user ID for the origin access identity, which you use when giving the origin access identity read permission to an object in Amazon S3.

### » **Using With CloudFront**

Normally, when referencing an origin access identity in CloudFront, you need to prefix the ID with the `origin-access-identity/cloudfront/` special path.

The `cloudfront_access_identity_path` allows this to be circumvented. The below snippet demonstrates use with the `s3_origin_config` structure for the `aws_cloudfront_web_distribution` resource:

```
s3_origin_config {
  origin_access_identity = "${aws_cloudfront_origin_access_identity.origin_access_identity.arn}"
}
```

## » Updating your bucket policy

Note that the AWS API may translate the `s3_canonical_user_id` CanonicalUser principal into an AWS IAM ARN principal when supplied in an `aws_s3_bucket` bucket policy, causing spurious diffs in Terraform. If you see this behaviour, use the `iam_arn` instead:

```
data "aws_iam_policy_document" "s3_policy" {
  statement {
    actions    = ["s3:GetObject"]
    resources  = ["${module.names.s3_endpoint_arn_base}/*"]

    principals {
      type       = "AWS"
      identifiers = ["${aws_cloudfront_origin_access_identity.origin_access_identity.iam_arn}"]
    }
  }
}

data "aws_iam_policy_document" "s3_policy" {
  statement {
    actions    = ["s3:ListBucket"]
    resources  = ["${module.names.s3_endpoint_arn_base}"]

    principals {
      type       = "AWS"
      identifiers = ["${aws_cloudfront_origin_access_identity.origin_access_identity.iam_arn}"]
    }
  }
}

resource "aws_s3_bucket" "bucket" {
  # ...
  policy = "${data.aws_iam_policy_document.s3_policy.json}"
}
```

## » Import

Cloudfront Origin Access Identities can be imported using the `id`, e.g.

```
$ terraform import aws_cloudfront_origin_access_identity.origin_access E74FTE3AEXAMPLE
```

## » **aws\_\_cloudtrail**

Provides a CloudTrail resource.

### » **Example Usage**

#### » **Basic**

Enable CloudTrail to capture all compatible management events in region. For capturing events from services like IAM, `include_global_service_events` must be enabled.

```
resource "aws_cloudtrail" "foobar" {
  name                        = "tf-trail-foobar"
  s3_bucket_name             = "${aws_s3_bucket.foo.id}"
  s3_key_prefix              = "prefix"
  include_global_service_events = false
}

resource "aws_s3_bucket" "foo" {
  bucket      = "tf-test-trail"
  force_destroy = true

  policy = <<POLICY
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AWSCloudTrailAclCheck",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudtrail.amazonaws.com"
      },
      "Action": "s3:GetBucketAcl",
      "Resource": "arn:aws:s3:::tf-test-trail"
    },
    {
      "Sid": "AWSCloudTrailWrite",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudtrail.amazonaws.com"
      },

```

```

        "Action": "s3:PutObject",
        "Resource": "arn:aws:s3:::tf-test-trail/*",
        "Condition": {
            "StringEquals": {
                "s3:x-amz-acl": "bucket-owner-full-control"
            }
        }
    }
]
}
POLICY
}

```

## » Data Event Logging

CloudTrail can log Data Events for certain services such as S3 bucket objects and Lambda function invocations. Additional information about data event configuration can be found in the CloudTrail API DataResource documentation.

## » Logging All Lambda Function Invocations

```

resource "aws_cloudtrail" "example" {
    # ... other configuration ...

    event_selector {
        read_write_type = "All"
        include_management_events = true

        data_resource {
            type    = "AWS::Lambda::Function"
            values = ["arn:aws:lambda"]
        }
    }
}

```

## » Logging All S3 Bucket Object Events

```

resource "aws_cloudtrail" "example" {
    # ... other configuration ...

    event_selector {
        read_write_type = "All"
        include_management_events = true

        data_resource {

```

```

        type    = "AWS::S3::Object"
        values = ["arn:aws:s3:::"]
    }
}
}

```

## » Logging Individual S3 Bucket Events

```

data "aws_s3_bucket" "important-bucket" {
    bucket = "important-bucket"
}

resource "aws_cloudtrail" "example" {
    # ... other configuration ...

    event_selector {
        read_write_type = "All"
        include_management_events = true

        data_resource {
            type    = "AWS::S3::Object"
            # Make sure to append a trailing '/' to your ARN if you want
            # to monitor all objects in a bucket.
            values = ["${data.aws_s3_bucket.important-bucket.arn}/"]
        }
    }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) Specifies the name of the trail.
- **s3\_bucket\_name** - (Required) Specifies the name of the S3 bucket designated for publishing log files.
- **s3\_key\_prefix** - (Optional) Specifies the S3 key prefix that precedes the name of the bucket you have designated for log file delivery.
- **cloud\_watch\_logs\_role\_arn** - (Optional) Specifies the role for the CloudWatch Logs endpoint to assume to write to a user's log group.
- **cloud\_watch\_logs\_group\_arn** - (Optional) Specifies a log group name using an Amazon Resource Name (ARN), that represents the log group to which CloudTrail logs will be delivered.
- **enable\_logging** - (Optional) Enables logging for the trail. Defaults to **true**. Setting this to **false** will pause logging.



- **include\_global\_service\_events** - (Optional) Specifies whether the trail is publishing events from global services such as IAM to the log files. Defaults to **true**.
- **is\_multi\_region\_trail** - (Optional) Specifies whether the trail is created in the current region or in all regions. Defaults to **false**.
- **sns\_topic\_name** - (Optional) Specifies the name of the Amazon SNS topic defined for notification of log file delivery.
- **enable\_log\_file\_validation** - (Optional) Specifies whether log file integrity validation is enabled. Defaults to **false**.
- **kms\_key\_id** - (Optional) Specifies the KMS key ARN to use to encrypt the logs delivered by CloudTrail.
- **event\_selector** - (Optional) Specifies an event selector for enabling data event logging. Fields documented below. Please note the CloudTrail limits when configuring these.
- **tags** - (Optional) A mapping of tags to assign to the trail

#### » Event Selector Arguments

For **event\_selector** the following attributes are supported.

- **read\_write\_type** (Optional) - Specify if you want your trail to log read-only events, write-only events, or all. By default, the value is All. You can specify only the following value: "ReadOnly", "WriteOnly", "All". Defaults to All.
- **include\_management\_events** (Optional) - Specify if you want your event selector to include management events for your trail.
- **data\_resource** (Optional) - Specifies logging data events. Fields documented below.

#### » Data Resource Arguments

For **data\_resource** the following attributes are supported.

- **type** (Required) - The resource type in which you want to log data events. You can specify only the following value: "AWS::S3::Object", "AWS::Lambda::Function"
- **values** (Required) - A list of ARN for the specified S3 buckets and object prefixes..

#### » Attribute Reference

The following attributes are exported:

- **id** - The name of the trail.
- **home\_region** - The region in which the trail was created.

- `arn` - The Amazon Resource Name of the trail.

## » Import

Cloudtrails can be imported using the `name`, e.g.

```
$ terraform import aws_cloudtrail.sample my-sample-trail
```

## » `aws_cloudwatch_dashboard`

Provides a CloudWatch Dashboard resource.

## » Example Usage

```
resource "aws_cloudwatch_dashboard" "main" {
  dashboard_name = "my-dashboard"
  dashboard_body = <<EOF
{
  "widgets": [
    {
      "type": "metric",
      "x": 0,
      "y": 0,
      "width": 12,
      "height": 6,
      "properties": {
        "metrics": [
          [
            "AWS/EC2",
            "CPUUtilization",
            "InstanceId",
            "i-012345"
          ]
        ],
        "period": 300,
        "stat": "Average",
        "region": "us-east-1",
        "title": "EC2 Instance CPU"
      }
    },
    {
      "type": "text",
      "x": 0,
```

```

        "y":7,
        "width":3,
        "height":3,
        "properties":{
            "markdown":"Hello world"
        }
    }
]
}
EOF
}

```

## » Argument Reference

The following arguments are supported:

- `dashboard_name` - (Required) The name of the dashboard.
- `dashboard_body` - (Required) The detailed information about the dashboard, including what widgets are included and their location on the dashboard. You can read more about the body structure in the documentation.

## » Attribute Reference

In addition to all arguments above, the following attributes are exported:

- `dashboard_arn` - The Amazon Resource Name (ARN) of the dashboard.

## » Import

CloudWatch dashboards can be imported using the `dashboard_name`, e.g.

```
$ terraform import aws_cloudwatch_dashboard.sample <dashboard_name>
```

## » `aws__cloudwatch__event__permission`

Provides a resource to create a CloudWatch Events permission to support cross-account events in the current account default event bus.

## » Example Usage

```
resource "aws_cloudwatch_event_permission" "DevAccountAccess" {
  principal = "123456789012"
```

```

    statement_id = "DevAccountAccess"
}

```

## » Argument Reference

The following arguments are supported:

- **principal** - (Required) The 12-digit AWS account ID that you are permitting to put events to your default event bus. Specify `*` to permit any account to put events to your default event bus.
- **statement\_id** - (Required) An identifier string for the external account that you are granting permissions to.
- **action** - (Optional) The action that you are enabling the other account to perform. Defaults to `events:PutEvents`.

## » Attributes Reference

The following additional attributes are exported:

- **id** - The statement ID of the CloudWatch Events permission.

## » Import

CloudWatch Events permissions can be imported using the statement ID, e.g.

```
$ terraform import aws_cloudwatch_event_permission.DevAccountAccess DevAccountAccess
```

## » aws\_\_cloudwatch\_\_event\_\_rule

Provides a CloudWatch Event Rule resource.

## » Example Usage

```

resource "aws_cloudwatch_event_rule" "console" {
  name           = "capture-aws-sign-in"
  description    = "Capture each AWS Console Sign In"

  event_pattern = <<PATTERN
{
  "detail-type": [
    "AWS Console Sign In via CloudTrail"
  ]
}

```

```

}
PATTERN
}

resource "aws_cloudwatch_event_target" "sns" {
  rule      = "${aws_cloudwatch_event_rule.console.name}"
  target_id = "SendToSNS"
  arn       = "${aws_sns_topic.aws_logins.arn}"
}

resource "aws_sns_topic" "aws_logins" {
  name = "aws-console-logins"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The rule's name. By default generated by Terraform.
- **name\_prefix** - (Optional) The rule's name. Conflicts with **name**.
- **schedule\_expression** - (Required, if **event\_pattern** isn't specified) The scheduling expression. For example, `cron(0 20 * * ? *)` or `rate(5 minutes)`.
- **event\_pattern** - (Required, if **schedule\_expression** isn't specified) Event pattern described a JSON object. See full documentation of CloudWatch Events and Event Patterns for details.
- **description** - (Optional) The description of the rule.
- **role\_arn** - (Optional) The Amazon Resource Name (ARN) associated with the role that is used for target invocation.
- **is\_enabled** - (Optional) Whether the rule should be enabled (defaults to `true`).

## » Attributes Reference

The following attributes are exported:

- **arn** - The Amazon Resource Name (ARN) of the rule.

## » Import

Cloudwatch Event Rules can be imported using the **name**, e.g.

```
$ terraform import aws_cloudwatch_event_rule.console capture-console-sign-in
```

## » `aws_cloudwatch_event_target`

Provides a CloudWatch Event Target resource.

### » Example Usage

```
resource "aws_cloudwatch_event_target" "yada" {
  target_id = "Yada"
  rule      = "${aws_cloudwatch_event_rule.console.name}"
  arn       = "${aws_kinesis_stream.test_stream.arn}"

  run_command_targets {
    key = "tag:Name"
    values = ["FooBar"]
  }

  run_command_targets {
    key = "InstanceIds"
    values = ["i-162058cd308bfffec2"]
  }
}

resource "aws_cloudwatch_event_rule" "console" {
  name          = "capture-ec2-scaling-events"
  description   = "Capture all EC2 scaling events"

  event_pattern = <<PATTERN
{
  "source": [
    "aws.autoscaling"
  ],
  "detail-type": [
    "EC2 Instance Launch Successful",
    "EC2 Instance Terminate Successful",
    "EC2 Instance Launch Unsuccessful",
    "EC2 Instance Terminate Unsuccessful"
  ]
}
PATTERN
}

resource "aws_kinesis_stream" "test_stream" {
  name          = "terraform-kinesis-test"
  shard_count   = 1
}
```

```
}
```

## » Example SSM Document Usage

```
data "aws_iam_policy_document" "ssm_lifecycle_trust" {
  statement {
    actions = ["sts:AssumeRole"]

    principals {
      type       = "Service"
      identifiers = ["events.amazonaws.com"]
    }
  }
}

data "aws_iam_policy_document" "ssm_lifecycle" {
  statement {
    effect   = "Allow"
    actions  = ["ssm:SendCommand"]
    resources = ["arn:aws:ec2:eu-west-1:1234567890:instance/*"]

    condition {
      test      = "StringEquals"
      variable  = "ec2:ResourceTag/Terminate"
      values    = ["*"]
    }
  }
}

statement {
  effect   = "Allow"
  actions  = ["ssm:SendCommand"]
  resources = ["${aws_ssm_document.stop_instance.arn}"]
}

resource "aws_iam_role" "ssm_lifecycle" {
  name               = "SSMLifecycle"
  assume_role_policy = "${data.aws_iam_policy_document.ssm_lifecycle_trust.json}"
}

resource "aws_iam_policy" "ssm_lifecycle" {
  name   = "SSMLifecycle"
  policy = "${data.aws_iam_policy_document.ssm_lifecycle.json}"
}
```

```

resource "aws_ssm_document" "stop_instance" {
  name          = "stop_instance"
  document_type = "Command"

  content = <<DOC
  {
    "schemaVersion": "1.2",
    "description": "Stop an instance",
    "parameters": {

    },
    "runtimeConfig": {
      "aws:runShellScript": {
        "properties": [
          {
            "id": "0.aws:runShellScript",
            "runCommand": ["halt"]
          }
        ]
      }
    }
  }
  DOC
}

resource "aws_cloudwatch_event_rule" "stop_instances" {
  name          = "StopInstance"
  description    = "Stop instances nightly"
  schedule_expression = "cron(0 0 * * ? *)"
}

resource "aws_cloudwatch_event_target" "stop_instances" {
  target_id = "StopInstance"
  arn       = "${aws_ssm_document.stop_instance.arn}"
  rule      = "${aws_cloudwatch_event_rule.stop_instances.name}"
  role_arn  = "${aws_iam_role.ssm_lifecycle.arn}"

  run_command_targets {
    key    = "tag:Terminate"
    values = ["midnight"]
  }
}

```



## » Example RunCommand Usage

```
resource "aws_cloudwatch_event_rule" "stop_instances" {
  name           = "StopInstance"
  description    = "Stop instances nightly"
  schedule_expression = "cron(0 0 * * ? *)"
}

resource "aws_cloudwatch_event_target" "stop_instances" {
  target_id = "StopInstance"
  arn       = "arn:aws:ssm:${var.aws_region}::document/AWS-RunShellScript"
  input     = "{\"commands\":[\"halt\"]}"
  rule      = "${aws_cloudwatch_event_rule.stop_instances.name}"
  role_arn  = "${aws_iam_role.ssm_lifecycle.arn}"

  run_command_targets {
    key    = "tag:Terminate"
    values = ["midnight"]
  }
}
```

## » Argument Reference

**Note:** `input` and `input_path` are mutually exclusive options.

**Note:** In order to be able to have your AWS Lambda function or SNS topic invoked by a CloudWatch Events rule, you must setup the right permissions using `aws_lambda_permission` or `aws_sns_topic.policy`. More info [here](#).

The following arguments are supported:

- **rule** - (Required) The name of the rule you want to add targets to.
- **target\_id** - (Optional) The unique target assignment ID. If missing, will generate a random, unique id.
- **arn** - (Required) The Amazon Resource Name (ARN) associated of the target.
- **input** - (Optional) Valid JSON text passed to the target.
- **input\_path** - (Optional) The value of the JSONPath that is used for extracting part of the matched event when passing it to the target.
- **role\_arn** - (Optional) The Amazon Resource Name (ARN) of the IAM role to be used for this target when the rule is triggered. Required if `ecs_target` is used.
- **run\_command\_targets** - (Optional) Parameters used when you are using the rule to invoke Amazon EC2 Run Command. Documented below. A maximum of 5 are allowed.

- **ecs\_target** - (Optional) Parameters used when you are using the rule to invoke Amazon ECS Task. Documented below. A maximum of 1 are allowed.
- **batch\_target** - (Optional) Parameters used when you are using the rule to invoke an Amazon Batch Job. Documented below. A maximum of 1 are allowed.
- **kinesis\_target** - (Optional) Parameters used when you are using the rule to invoke an Amazon Kinesis Stream. Documented below. A maximum of 1 are allowed.
- **sqs\_target** - (Optional) Parameters used when you are using the rule to invoke an Amazon SQS Queue. Documented below. A maximum of 1 are allowed.
- **input\_transformer** - (Optional) Parameters used when you are providing a custom input to a target based on certain event data.

**run\_command\_targets** support the following:

- **key** - (Required) Can be either **tag:tag-key** or **InstanceIds**.
- **values** - (Required) If Key is **tag:tag-key**, Values is a list of tag values. If Key is **InstanceIds**, Values is a list of Amazon EC2 instance IDs.

**ecs\_target** support the following:

- **task\_count** - (Optional) The number of tasks to create based on the TaskDefinition. The default is 1.
- **task\_definition\_arn** - (Required) The ARN of the task definition to use if the event target is an Amazon ECS cluster.

**batch\_target** support the following:

- **job\_definition** - (Required) The ARN or name of the job definition to use if the event target is an AWS Batch job. This job definition must already exist.
- **job\_name** - (Required) The name to use for this execution of the job, if the target is an AWS Batch job.
- **array\_size** - (Optional) The size of the array, if this is an array batch job. Valid values are integers between 2 and 10,000.
- **job\_attempts** - (Optional) The number of times to attempt to retry, if the job fails. Valid values are 1 to 10.

**kinesis\_target** support the following:

- **partition\_key\_path** - (Optional) The JSON path to be extracted from the event and used as the partition key.

**sqs\_target** support the following:

- **message\_group\_id** - (Optional) The FIFO message group ID to use as the target.

**input\_transformer** support the following:

- **input\_paths** - (Optional) Key value pairs specified in the form of JSON-Path (for example, time = \$.time)
- **input\_template** - (Required) Structure containing the template body.

## » **aws\_\_cloudwatch\_\_log\_\_destination**

Provides a CloudWatch Logs destination resource.

### » **Example Usage**

```
resource "aws_cloudwatch_log_destination" "test_destination" {
  name      = "test_destination"
  role_arn  = "${aws_iam_role.iam_for_cloudwatch.arn}"
  target_arn = "${aws_kinesis_stream.kinesis_for_cloudwatch.arn}"
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) A name for the log destination
- **role\_arn** - (Required) The ARN of an IAM role that grants Amazon CloudWatch Logs permissions to put data into the target
- **target\_arn** - (Required) The ARN of the target Amazon Kinesis stream or Amazon Lambda resource for the destination

### » **Attributes Reference**

The following attributes are exported:

- **arn** - The Amazon Resource Name (ARN) specifying the log destination.

### » **Import**

CloudWatch Logs destinations can be imported using the **name**, e.g.

```
$ terraform import aws_cloudwatch_log_destination.test_destination test_destination
```

## » **aws\_\_cloudwatch\_\_log\_\_destination\_\_policy**

Provides a CloudWatch Logs destination policy resource.

## » Example Usage

```
resource "aws_cloudwatch_log_destination" "test_destination" {
  name          = "test_destination"
  role_arn      = "${aws_iam_role.iam_for_cloudwatch.arn}"
  target_arn    = "${aws_kinesis_stream.kinesis_for_cloudwatch.arn}"
}

data "aws_iam_policy_document" "test_destination_policy" {
  statement {
    effect = "Allow"

    principals = {
      type = "AWS"

      identifiers = [
        "123456789012",
      ]
    }

    actions = [
      "logs:PutSubscriptionFilter",
    ]

    resources = [
      "${aws_cloudwatch_log_destination.test_destination.arn}",
    ]
  }
}

resource "aws_cloudwatch_log_destination_policy" "test_destination_policy" {
  destination_name = "${aws_cloudwatch_log_destination.test_destination.name}"
  access_policy    = "${data.aws_iam_policy_document.test_destination_policy.json}"
}
```

## » Argument Reference

The following arguments are supported:

- **destination\_name** - (Required) A name for the subscription filter
- **access\_policy** - (Required) The policy document. This is a JSON formatted string.

## » Import

CloudWatch Logs destination policies can be imported using the `destination_name`, e.g.

```
$ terraform import aws_cloudwatch_log_destination_policy.test_destination_policy test_destination_name
```

## » `aws_cloudwatch_log_group`

Provides a CloudWatch Log Group resource.

## » Example Usage

```
resource "aws_cloudwatch_log_group" "yada" {
  name = "Yada"

  tags {
    Environment = "production"
    Application = "serviceA"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional, Forces new resource) The name of the log group. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **retention\_in\_days** - (Optional) Specifies the number of days you want to retain log events in the specified log group.
- **kms\_key\_id** - (Optional) The ARN of the KMS Key to use when encrypting log data. Please note, after the AWS KMS CMK is disassociated from the log group, AWS CloudWatch Logs stops encrypting newly ingested data for the log group. All previously ingested data remains encrypted, and AWS CloudWatch Logs requires permissions for the CMK whenever the encrypted data is requested.
- **tags** - (Optional) A mapping of tags to assign to the resource.

## » Attributes Reference

The following attributes are exported:

- **arn** - The Amazon Resource Name (ARN) specifying the log group.

## » Import

Cloudwatch Log Groups can be imported using the **name**, e.g.

```
$ terraform import aws_cloudwatch_log_group.test_group yada
```

## » `aws_cloudwatch_log_metric_filter`

Provides a CloudWatch Log Metric Filter resource.

## » Example Usage

```
resource "aws_cloudwatch_log_metric_filter" "yada" {
  name          = "MyAppAccessCount"
  pattern       = ""
  log_group_name = "${aws_cloudwatch_log_group.dada.name}"

  metric_transformation {
    name      = "EventCount"
    namespace = "YourNamespace"
    value     = "1"
  }
}

resource "aws_cloudwatch_log_group" "dada" {
  name = "MyApp/access.log"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A name for the metric filter.
- **pattern** - (Required) A valid CloudWatch Logs filter pattern for extracting metric data out of ingested log events.
- **log\_group\_name** - (Required) The name of the log group to associate the metric filter with.
- **metric\_transformation** - (Required) A block defining collection of information needed to define how metric data gets emitted. See below.

The **metric\_transformation** block supports the following arguments:

- **name** - (Required) The name of the CloudWatch metric to which the monitored log information should be published (e.g. **ErrorCount**)
- **namespace** - (Required) The destination namespace of the CloudWatch metric.
- **value** - (Required) What to publish to the metric. For example, if you're counting the occurrences of a particular term like "Error", the value will be "1" for each occurrence. If you're counting the bytes transferred the published value will be the value in the log event.
- **default\_value** - (Optional) The value to emit when a filter pattern does not match a log event.

## » Attributes Reference

The following attributes are exported:

- **id** - The name of the metric filter.

## » `aws_cloudwatch_log_resource_policy`

Provides a resource to manage a CloudWatch log resource policy.

## » Example Usage

### » Route53 Query Logging

```
data "aws_iam_policy_document" "route53-query-logging-policy" {
  statement {
    actions = [
      "logs:CreateLogStream",
      "logs:PutLogEvents",
    ]

    resources = ["arn:aws:logs:*:*:log-group:/aws/route53/*"]

    principals {
      identifiers = ["route53.amazonaws.com"]
      type       = "Service"
    }
  }
}

resource "aws_cloudwatch_log_resource_policy" "route53-query-logging-policy" {
  policy_document = "${data.aws_iam_policy_document.route53-query-logging-policy.json}"
}
```

```

    policy_name      = "route53-query-logging-policy"
  }

```

## » Argument Reference

The following arguments are supported:

- `policy_document` - (Required) Details of the resource policy, including the identity of the principal that is enabled to put logs to this account. This is formatted as a JSON string. Maximum length of 5120 characters.
- `policy_name` - (Required) Name of the resource policy.

## » Attributes Reference

The following additional attributes are exported:

- `id` - The name of the CloudWatch log resource policy

## » Import

CloudWatch log resource policies can be imported using the policy name, e.g.

```
$ terraform import aws_cloudwatch_log_resource_policy.MyPolicy MyPolicy
```

## » `aws__cloudwatch__log__stream`

Provides a CloudWatch Log Stream resource.

## » Example Usage

```

resource "aws_cloudwatch_log_group" "yada" {
  name = "Yada"
}

resource "aws_cloudwatch_log_stream" "foo" {
  name                = "SampleLogStream1234"
  log_group_name      = "${aws_cloudwatch_log_group.yada.name}"
}

```



## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the log stream. Must not be longer than 512 characters and must not contain :
- **log\_group\_name** - (Required) The name of the log group under which the log stream is to be created.

## » Attributes Reference

The following attributes are exported:

- **arn** - The Amazon Resource Name (ARN) specifying the log stream.

## » `aws_cloudwatch_log_subscription_filter`

Provides a CloudWatch Logs subscription filter resource.

## » Example Usage

```
resource "aws_cloudwatch_log_subscription_filter" "test_lambdafunction_logfilter" {
  name          = "test_lambdafunction_logfilter"
  role_arn      = "${aws_iam_role.iam_for_lambda.arn}"
  log_group_name = "/aws/lambda/example_lambda_name"
  filter_pattern = "logtype test"
  destination_arn = "${aws_kinesis_stream.test_logstream.arn}"
  distribution   = "Random"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A name for the subscription filter
- **destination\_arn** - (Required) The ARN of the destination to deliver matching log events to. Kinesis stream or Lambda function ARN.
- **filter\_pattern** - (Required) A valid CloudWatch Logs filter pattern for subscribing to a filtered stream of log events.
- **log\_group\_name** - (Required) The name of the log group to associate the subscription filter with

- **role\_arn** - (Optional) The ARN of an IAM role that grants Amazon CloudWatch Logs permissions to deliver ingested log events to the destination. If you use Lambda as a destination, you should skip this argument and use **aws\_lambda\_permission** resource for granting access from CloudWatch logs to the destination Lambda function.
- **distribution** - (Optional) The method used to distribute log data to the destination. By default log data is grouped by log stream, but the grouping can be set to random for a more even distribution. This property is only applicable when the destination is an Amazon Kinesis stream. Valid values are "Random" and "ByLogStream".

## » Attributes Reference

No extra attributes are exported.

## » `aws_cloudwatch_metric_alarm`

Provides a CloudWatch Metric Alarm resource.

## » Example Usage

```
resource "aws_cloudwatch_metric_alarm" "foobar" {
  alarm_name          = "terraform-test-foobar5"
  comparison_operator = "GreaterThanOrEqualToThreshold"
  evaluation_periods  = "2"
  metric_name         = "CPUUtilization"
  namespace           = "AWS/EC2"
  period              = "120"
  statistic            = "Average"
  threshold            = "80"
  alarm_description   = "This metric monitors ec2 cpu utilization"
  insufficient_data_actions = []
}
```

## » Example in Conjunction with Scaling Policies

```
resource "aws_autoscaling_policy" "bat" {
  name                = "foobar3-terraform-test"
  scaling_adjustment  = 4
  adjustment_type     = "ChangeInCapacity"
  cooldown             = 300
}
```

```

    autoscaling_group_name = "${aws_autoscaling_group.bar.name}"
}

resource "aws_cloudwatch_metric_alarm" "bat" {
    alarm_name           = "terraform-test-foobar5"
    comparison_operator = "GreaterThanOrEqualToThreshold"
    evaluation_periods   = "2"
    metric_name          = "CPUUtilization"
    namespace            = "AWS/EC2"
    period               = "120"
    statistic             = "Average"
    threshold            = "80"

    dimensions {
        AutoScalingGroupName = "${aws_autoscaling_group.bar.name}"
    }

    alarm_description = "This metric monitors ec2 cpu utilization"
    alarm_actions     = ["${aws_autoscaling_policy.bat.arn}"]
}

```

**NOTE:** You cannot create a metric alarm consisting of both `statistic` and `extended_statistic` parameters. You must choose one or the other

## » Argument Reference

See related part of AWS Docs for details about valid values.

The following arguments are supported:

- **alarm\_name** - (Required) The descriptive name for the alarm. This name must be unique within the user's AWS account
- **comparison\_operator** - (Required) The arithmetic operation to use when comparing the specified Statistic and Threshold. The specified Statistic value is used as the first operand. Either of the following is supported: `GreaterThanOrEqualToThreshold`, `GreaterThanThreshold`, `LessThanThreshold`, `LessThanOrEqualToThreshold`.
- **evaluation\_periods** - (Required) The number of periods over which data is compared to the specified threshold.
- **metric\_name** - (Required) The name for the alarm's associated metric. See docs for supported metrics.
- **namespace** - (Required) The namespace for the alarm's associated metric. See docs for the list of namespaces. See docs for supported metrics.
- **period** - (Required) The period in seconds over which the specified `statistic` is applied.
- **statistic** - (Optional) The statistic to apply to the alarm's associated

metric. Either of the following is supported: `SampleCount`, `Average`, `Sum`, `Minimum`, `Maximum`

- **threshold** - (Required) The value against which the specified statistic is compared.
- **actions\_enabled** - (Optional) Indicates whether or not actions should be executed during any changes to the alarm's state. Defaults to `true`.
- **alarm\_actions** - (Optional) The list of actions to execute when this alarm transitions into an ALARM state from any other state. Each action is specified as an Amazon Resource Number (ARN).
- **alarm\_description** - (Optional) The description for the alarm.
- **datapoints\_to\_alarm** - (Optional) The number of datapoints that must be breaching to trigger the alarm.
- **dimensions** - (Optional) The dimensions for the alarm's associated metric. For the list of available dimensions see the AWS documentation [here](#).
- **insufficient\_data\_actions** - (Optional) The list of actions to execute when this alarm transitions into an INSUFFICIENT\_DATA state from any other state. Each action is specified as an Amazon Resource Number (ARN).
- **ok\_actions** - (Optional) The list of actions to execute when this alarm transitions into an OK state from any other state. Each action is specified as an Amazon Resource Number (ARN).
- **unit** - (Optional) The unit for the alarm's associated metric.
- **extended\_statistic** - (Optional) The percentile statistic for the metric associated with the alarm. Specify a value between p0.0 and p100.
- **treat\_missing\_data** - (Optional) Sets how this alarm is to handle missing data points. The following values are supported: `missing`, `ignore`, `breaching` and `notBreaching`. Defaults to `missing`.
- **evaluate\_low\_sample\_count\_percentiles** - (Optional) Used only for alarms based on percentiles. If you specify `ignore`, the alarm state will not change during periods with too few data points to be statistically significant. If you specify `evaluate` or omit this parameter, the alarm will always be evaluated and possibly change state no matter how many data points are available. The following values are supported: `ignore`, and `evaluate`.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the health check

## » Import

Cloud Metric Alarms can be imported using the `alarm_name`, e.g.

```
$ terraform import aws_cloudwatch_metric_alarm.test alarm-12345
```

## » `aws__codebuild__project`

Provides a CodeBuild Project resource.

### » Example Usage

```
resource "aws_s3_bucket" "foo" {
  bucket = "test-bucket"
  acl    = "private"
}

resource "aws_iam_role" "codebuild_role" {
  name = "codebuild-role-"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "codebuild.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF
}

resource "aws_iam_policy" "codebuild_policy" {
  name       = "codebuild-policy"
  path       = "/service-role/"
  description = "Policy used in trust relationship with CodeBuild"

  policy = <<POLICY
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```

        "Resource": [
            "*"
        ],
        "Action": [
            "logs:CreateLogGroup",
            "logs:CreateLogStream",
            "logs:PutLogEvents"
        ]
    }
}
POLICY
}

resource "aws_iam_policy_attachment" "codebuild_policy_attachment" {
    name          = "codebuild-policy-attachment"
    policy_arn    = "${aws_iam_policy.codebuild_policy.arn}"
    roles         = ["${aws_iam_role.codebuild_role.id}"]
}

resource "aws_codebuild_project" "foo" {
    name          = "test-project"
    description   = "test_codebuild_project"
    build_timeout = "5"
    service_role = "${aws_iam_role.codebuild_role.arn}"

    artifacts {
        type = "NO_ARTIFACTS"
    }

    cache {
        type      = "S3"
        location = "${aws_s3_bucket.foo.bucket}"
    }

    environment {
        compute_type = "BUILD_GENERAL1_SMALL"
        image        = "aws/codebuild/nodejs:6.3.1"
        type         = "LINUX_CONTAINER"

        environment_variable {
            "name"  = "SOME_KEY1"
            "value" = "SOME_VALUE1"
        }

        environment_variable {

```

```

        "name" = "SOME_KEY2"
        "value" = "SOME_VALUE2"
    }
}

source {
    type = "GITHUB"
    location = "https://github.com/mitchellh/packer.git"
}

vpc_config {
    vpc_id = "vpc-725fca"

    subnets = [
        "subnet-ba35d2e0",
        "subnet-ab129af1",
    ]

    security_group_ids = [
        "sg-f9f27d91",
        "sg-e4f48g23",
    ]
}

tags {
    "Environment" = "Test"
}
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The projects name.
- **description** - (Optional) A short description of the project.
- **encryption\_key** - (Optional) The AWS Key Management Service (AWS KMS) customer master key (CMK) to be used for encrypting the build project's build output artifacts.
- **service\_role** - (Optional) The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that enables AWS CodeBuild to interact with dependent AWS services on behalf of the AWS account.
- **build\_timeout** - (Optional) How long in minutes, from 5 to 480 (8 hours), for AWS CodeBuild to wait until timing out any related build that does not get marked as completed. The default is 60 minutes.

- **tags** - (Optional) A mapping of tags to assign to the resource.
- **artifacts** - (Required) Information about the project's build output artifacts. Artifact blocks are documented below.
- **cache** - (Optional) Information about the cache storage for the project. Cache blocks are documented below.
- **environment** - (Required) Information about the project's build environment. Environment blocks are documented below.
- **source** - (Required) Information about the project's input source code. Source blocks are documented below.
- **vpc\_config** - (Optional) Configuration for the builds to run inside a VPC. VPC config blocks are documented below.

**artifacts** supports the following:

- **type** - (Required) The build output artifact's type. Valid values for this parameter are: `CODEPIPELINE`, `NO_ARTIFACTS` or `S3`.
- **location** - (Optional) Information about the build output artifact location. If **type** is set to `CODEPIPELINE` or `NO_ARTIFACTS` then this value will be ignored. If **type** is set to `S3`, this is the name of the output bucket. If **path** is not also specified, then **location** can also specify the path of the output artifact in the output bucket.
- **name** - (Optional) The name of the project. If **type** is set to `S3`, this is the name of the output artifact object
- **namespace\_type** - (Optional) The namespace to use in storing build artifacts. If **type** is set to `S3`, then valid values for this parameter are: `BUILD_ID` or `NONE`.
- **packaging** - (Optional) The type of build output artifact to create. If **type** is set to `S3`, valid values for this parameter are: `NONE` or `ZIP`
- **path** - (Optional) If **type** is set to `S3`, this is the path to the output artifact

**cache** supports the following:

- **type** - (Optional) The type of storage that will be used for the AWS CodeBuild project cache. Valid values: `NO_CACHE` and `S3`. Defaults to `NO_CACHE`.
- **location** - (Required when cache type is `S3`) The location where the AWS CodeBuild project stores cached resources. For type `S3` the value must be a valid S3 bucket name/prefix.

**environment** supports the following:

- **compute\_type** - (Required) Information about the compute resources the build project will use. Available values for this parameter are: `BUILD_GENERAL1_SMALL`, `BUILD_GENERAL1_MEDIUM` or `BUILD_GENERAL1_LARGE`
- **image** - (Required) The *image identifier* of the Docker image to use for this build project (list of Docker images provided by AWS CodeBuild.). You can read more about the AWS curated environment images in the documentation.



- **type** - (Required) The type of build environment to use for related builds. The only valid value is `LINUX_CONTAINER`.
- **privileged\_mode** - (Optional) If set to true, enables running the Docker daemon inside a Docker container. Defaults to `false`.
- **environment\_variable** - (Optional) A set of environment variables to make available to builds for this build project.

**environment\_variable** supports the following:

- **name** - (Required) The environment variable's name or key.
- **value** - (Required) The environment variable's value.

**source** supports the following:

- **type** - (Required) The type of repository that contains the source code to be built. Valid values for this parameter are: `CODECOMMIT`, `CODEPIPELINE`, `GITHUB`, `GITHUB_ENTERPRISE`, `BITBUCKET` or `S3`.
- **auth** - (Optional) Information about the authorization settings for AWS CodeBuild to access the source code to be built. Auth blocks are documented below.
- **buildspec** - (Optional) The build spec declaration to use for this build project's related builds.
- **location** - (Optional) The location of the source code from git or s3.

**auth** supports the following:

- **type** - (Required) The authorization type to use. The only valid value is `OAUTH`.
- **resource** - (Optional) The resource value that applies to the specified authorization type.

**vpc\_config** supports the following:

- **vpc\_id** - (Required) The ID of the VPC within which to run builds.
- **subnets** - (Required) The subnet IDs within which to run builds.
- **security\_group\_ids** - (Required) The security group IDs to assign to running builds.

## » Attributes Reference

The following attributes are exported:

- **id** - The ARN of the CodeBuild project.
- **description** - A short description of the project.
- **encryption\_key** - The AWS Key Management Service (AWS KMS) customer master key (CMK) that was used for encrypting the build project's build output artifacts.
- **name** - The project's name.
- **service\_role** - The ARN of the IAM service role.

## » **aws\_\_codecommit\_repository**

Provides a CodeCommit Repository Resource.

**NOTE on CodeCommit Availability:** The CodeCommit is not yet rolled out in all regions - available regions are listed the AWS Docs.

### » **Example Usage**

```
resource "aws_codecommit_repository" "test" {
  repository_name = "MyTestRepository"
  description     = "This is the Sample App Repository"
}
```

### » **Argument Reference**

The following arguments are supported:

- **repository\_name** - (Required) The name for the repository. This needs to be less than 100 characters.
- **description** - (Optional) The description of the repository. This needs to be less than 1000 characters
- **default\_branch** - (Optional) The default branch of the repository. The branch specified here needs to exist.

### » **Attributes Reference**

The following attributes are exported:

- **repository\_id** - The ID of the repository
- **arn** - The ARN of the repository
- **clone\_url\_http** - The URL to use for cloning the repository over HTTPS.
- **clone\_url\_ssh** - The URL to use for cloning the repository over SSH.

### » **Import**

Codecommit repository can be imported using repository name, e.g.

```
$ terraform import aws_codecommit_repository.imported ExistingRepo
```

## » `aws__codecommit__trigger`

Provides a CodeCommit Trigger Resource.

**NOTE on CodeCommit:** The CodeCommit is not yet rolled out in all regions - available regions are listed the AWS Docs.

### » Example Usage

```
resource "aws_codecommit_trigger" "test" {
  depends_on      = ["aws_codecommit_repository.test"]
  repository_name = "my_test_repository"

  trigger {
    name           = "noname"
    events         = ["all"]
    destination_arn = "${aws_sns_topic.test.arn}"
  }
}
```

### » Argument Reference

The following arguments are supported:

- **repository\_name** - (Required) The name for the repository. This needs to be less than 100 characters.
- **name** - (Required) The name of the trigger.
- **destination\_arn** - (Required) The ARN of the resource that is the target for a trigger. For example, the ARN of a topic in Amazon Simple Notification Service (SNS).
- **custom\_data** - (Optional) Any custom data associated with the trigger that will be included in the information sent to the target of the trigger.
- **branches** - (Optional) The branches that will be included in the trigger configuration. If no branches are specified, the trigger will apply to all branches.
- **events** - (Required) The repository events that will cause the trigger to run actions in another service, such as sending a notification through Amazon Simple Notification Service (SNS). If no events are specified, the trigger will run for all repository events. Event types include: `all`, `updateReference`, `createReference`, `deleteReference`.

## » **aws\_\_codedeploy\_\_app**

Provides a CodeDeploy application to be used as a basis for deployments

### » **Example Usage**

```
resource "aws_codedeploy_app" "foo" {  
  name = "foo"  
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) The name of the application.

### » **Attribute Reference**

The following arguments are exported:

- **id** - Amazon's assigned ID for the application.
- **name** - The application's name.

## » **aws\_\_codedeploy\_\_deployment\_\_config**

Provides a CodeDeploy deployment config for an application

### » **Example Usage**

```
resource "aws_codedeploy_deployment_config" "foo" {  
  deployment_config_name = "test-deployment-config"  
  
  minimum_healthy_hosts {  
    type = "HOST_COUNT"  
    value = 2  
  }  
}
```

```
resource "aws_codedeploy_deployment_group" "foo" {  
  app_name          = "${aws_codedeploy_app.foo_app.name}"  
  deployment_group_name = "bar"
```

```

service_role_arn      = "${aws_iam_role.foo_role.arn}"
deployment_config_name = "${aws_codedeploy_deployment_config.foo.id}"

ec2_tag_filter {
  key   = "filterkey"
  type  = "KEY_AND_VALUE"
  value = "filtervalue"
}

trigger_configuration {
  trigger_events      = ["DeploymentFailure"]
  trigger_name        = "foo-trigger"
  trigger_target_arn = "foo-topic-arn"
}

auto_rollback_configuration {
  enabled = true
  events  = ["DEPLOYMENT_FAILURE"]
}

alarm_configuration {
  alarms = ["my-alarm-name"]
  enabled = true
}
}

```

## » Argument Reference

The following arguments are supported:

- `deployment_config_name` - (Required) The name of the deployment config.
- `minimum_healthy_hosts` - (Optional) A `minimum_healthy_hosts` block. Minimum Healthy Hosts are documented below.

A `minimum_healthy_hosts` block support the following:

- `type` - (Required) The type can either be `FLEET_PERCENT` or `HOST_COUNT`.
- `value` - (Required) The value when the type is `FLEET_PERCENT` represents the minimum number of healthy instances as a percentage of the total number of instances in the deployment. If you specify `FLEET_PERCENT`, at the start of the deployment, AWS CodeDeploy converts the percentage to the equivalent number of instance and rounds up fractional instances. When the type is `HOST_COUNT`, the value represents the minimum number of healthy instances as an absolute value.

## » Attributes Reference

The following attributes are exported:

- `id` - The deployment group's config name.
- `deployment_config_id` - The AWS Assigned deployment config id

## » `aws_codedeploy_deployment_group`

Provides a CodeDeploy Deployment Group for a CodeDeploy Application

## » Example Usage

```
resource "aws_iam_role" "example" {
  name = "example-role"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "codedeploy.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF
}

resource "aws_iam_role_policy" "example" {
  name = "example-policy"
  role = "${aws_iam_role.example.id}"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
```

```

        "autoscaling:CompleteLifecycleAction",
        "autoscaling>DeleteLifecycleHook",
        "autoscaling:DescribeAutoScalingGroups",
        "autoscaling:DescribeLifecycleHooks",
        "autoscaling:PutLifecycleHook",
        "autoscaling:RecordLifecycleActionHeartbeat",
        "codedeploy:*",
        "ec2:DescribeInstances",
        "ec2:DescribeInstanceStatus",
        "tag:GetTags",
        "tag:GetResources",
        "sns:Publish"
    ],
    "Resource": "*"
}
]
}
EOF
}

resource "aws_codedeploy_app" "example" {
    name = "example-app"
}

resource "aws_sns_topic" "example" {
    name = "example-topic"
}

resource "aws_codedeploy_deployment_group" "example" {
    app_name          = "${aws_codedeploy_app.example.name}"
    deployment_group_name = "example-group"
    service_role_arn   = "${aws_iam_role.example.arn}"

    ec2_tag_set {
        ec2_tag_filter {
            key   = "filterkey1"
            type  = "KEY_AND_VALUE"
            value = "filtervalue"
        }

        ec2_tag_filter {
            key   = "filterkey2"
            type  = "KEY_AND_VALUE"
            value = "filtervalue"
        }
    }
}

```

```

trigger_configuration {
  trigger_events      = ["DeploymentFailure"]
  trigger_name        = "example-trigger"
  trigger_target_arn = "${aws_sns_topic.example.arn}"
}

auto_rollback_configuration {
  enabled = true
  events  = ["DEPLOYMENT_FAILURE"]
}

alarm_configuration {
  alarms  = ["my-alarm-name"]
  enabled = true
}
}

```

## » Using Blue Green Deployments

```

resource "aws_codedeploy_app" "example" {
  name = "example-app"
}

resource "aws_codedeploy_deployment_group" "example" {
  app_name                = "${aws_codedeploy_app.example.name}"
  deployment_group_name   = "example-group"
  service_role_arn        = "${aws_iam_role.example.arn}"

  deployment_style {
    deployment_option = "WITH_TRAFFIC_CONTROL"
    deployment_type   = "BLUE_GREEN"
  }

  load_balancer_info {
    elb_info {
      name = "example-elb"
    }
  }

  blue_green_deployment_config {
    deployment_ready_option {
      action_on_timeout = "STOP_DEPLOYMENT"
      wait_time_in_minutes = 60
    }
  }
}

```



```

    green_fleet_provisioning_option {
        action = "DISCOVER_EXISTING"
    }

    terminate_blue_instances_on_deployment_success {
        action = "KEEP_ALIVE"
    }
}

```

## » Argument Reference

The following arguments are supported:

- **app\_name** - (Required) The name of the application.
- **deployment\_group\_name** - (Required) The name of the deployment group.
- **service\_role\_arn** - (Required) The service role ARN that allows deployments.
- **autoscaling\_groups** - (Optional) Autoscaling groups associated with the deployment group.
- **deployment\_config\_name** - (Optional) The name of the group's deployment config. The default is "CodeDeployDefault.OneAtATime".
- **ec2\_tag\_filter** - (Optional) Tag filters associated with the deployment group. See the AWS docs for details.
- **ec2\_tag\_set** - (Optional) Sets of Tag filters associated with the deployment group, which are referred to as *tag groups* in the document. See the AWS docs for details.
- **on\_premises\_instance\_tag\_filter** - (Optional) On premise tag filters associated with the group. See the AWS docs for details.
- **trigger\_configuration** - (Optional) Trigger Configurations for the deployment group (documented below).
- **auto\_rollback\_configuration** - (Optional) The automatic rollback configuration associated with the deployment group (documented below).
- **alarm\_configuration** - (Optional) Information about alarms associated with the deployment group (documented below).
- **deployment\_style** - (Optional) Information about the type of deployment, either in-place or blue/green, you want to run and whether to route deployment traffic behind a load balancer (documented below).
- **load\_balancer\_info** - (Optional) Information about the load balancer to use in a blue/green deployment (documented below).
- **blue\_green\_deployment\_config** - (Optional) Information about blue/green deployment options for a deployment group (documented below).

## » Tag Filters

Both `ec2_tag_filter` and `on_premises_tag_filter` support the following:

- **key** - (Optional) The key of the tag filter.
- **type** - (Optional) The type of the tag filter, either `KEY_ONLY`, `VALUE_ONLY`, or `KEY_AND_VALUE`.
- **value** - (Optional) The value of the tag filter.

Multiple occurrences of `ec2_tag_filter` are allowed, where any instance that matches to at least one of the tag filters is selected.

## » Tag Groups

You can form a tag group by putting a set of tag filters into `ec2_tag_set`. If multiple tag groups are specified, any instance that matches to at least one tag filter of every tag group is selected.

## » Trigger Configuration

Add triggers to a Deployment Group to receive notifications about events related to deployments or instances in the group. Notifications are sent to subscribers of the **SNS** topic associated with the trigger. *CodeDeploy must have permission to publish to the topic from this deployment group.* `trigger_configuration` supports the following:

- **trigger\_events** - (Required) The event type or types for which notifications are triggered. Some values that are supported: `DeploymentStart`, `DeploymentSuccess`, `DeploymentFailure`, `DeploymentStop`, `DeploymentRollback`, `InstanceStart`, `InstanceSuccess`, `InstanceFailure`. See the CodeDeploy documentation for all possible values.
- **trigger\_name** - (Required) The name of the notification trigger.
- **trigger\_target\_arn** - (Required) The ARN of the SNS topic through which notifications are sent.

## » Auto Rollback Configuration

You can configure a deployment group to automatically rollback when a deployment fails or when a monitoring threshold you specify is met. In this case, the last known good version of an application revision is deployed. `auto_rollback_configuration` supports the following:

- **enabled** - (Optional) Indicates whether a defined automatic rollback configuration is currently enabled for this Deployment Group. If you enable automatic rollback, you must specify at least one event type.

- **events** - (Optional) The event type or types that trigger a rollback. Supported types are `DEPLOYMENT_FAILURE` and `DEPLOYMENT_STOP_ON_ALARM`.

*Only one 'auto\_rollback' configuration is allowed.*

## » Alarm Configuration

You can configure a deployment to stop when a **CloudWatch** alarm detects that a metric has fallen below or exceeded a defined threshold. `alarm_configuration` supports the following:

- **alarms** - (Optional) A list of alarms configured for the deployment group. *A maximum of 10 alarms can be added to a deployment group.*
- **enabled** - (Optional) Indicates whether the alarm configuration is enabled. This option is useful when you want to temporarily deactivate alarm monitoring for a deployment group without having to add the same alarms again later.
- **ignore\_poll\_alarm\_failure** - (Optional) Indicates whether a deployment should continue if information about the current state of alarms cannot be retrieved from CloudWatch. The default value is **false**.
  - **true**: The deployment will proceed even if alarm status information can't be retrieved.
  - **false**: The deployment will stop if alarm status information can't be retrieved.

*Only one alarm\_configuration is allowed.*

## » Deployment Style

You can configure the type of deployment, either in-place or blue/green, you want to run and whether to route deployment traffic behind a load balancer. `deployment_style` supports the following:

- **deployment\_option** - (Optional) Indicates whether to route deployment traffic behind a load balancer. Valid Values are `WITH_TRAFFIC_CONTROL` or `WITHOUT_TRAFFIC_CONTROL`.
- **deployment\_type** - (Optional) Indicates whether to run an in-place deployment or a blue/green deployment. Valid Values are `IN_PLACE` or `BLUE_GREEN`.

*Only one deployment\_style is allowed.*

## » Load Balancer Info

You can configure the **Load Balancer** to use in a deployment. `load_balancer_info` supports the following:

- **elb\_info** - (Optional) The load balancer to use in a deployment.
- **target\_group\_info** - (Optional) The target group to use in a deployment.

Only one **load\_balancer\_info** is supported per deployment group.

**elb\_info** supports the following:

- **name** - (Optional) The name of the load balancer that will be used to route traffic from original instances to replacement instances in a blue/green deployment. For in-place deployments, the name of the load balancer that instances are deregistered from so they are not serving traffic during a deployment, and then re-registered with after the deployment completes.

**target\_group\_info** supports the following:

- **name** - (Optional) The name of the target group that instances in the original environment are deregistered from, and instances in the replacement environment registered with. For in-place deployments, the name of the target group that instances are deregistered from, so they are not serving traffic during a deployment, and then re-registered with after the deployment completes.

*Only a single **elb\_info** or **target\_group\_info** can be used in a deployment.*

## » Blue Green Deployment Configuration

You can configure options for a blue/green deployment. **blue\_green\_deployment\_config** supports the following:

- **deployment\_ready\_option** - (Optional) Information about the action to take when newly provisioned instances are ready to receive traffic in a blue/green deployment (documented below).
- **green\_fleet\_provisioning\_option** - (Optional) Information about how instances are provisioned for a replacement environment in a blue/green deployment (documented below).
- **terminate\_blue\_instances\_on\_deployment\_success** - (Optional) Information about whether to terminate instances in the original fleet during a blue/green deployment (documented below).

*Only one **blue\_green\_deployment\_config** is allowed.*

You can configure how traffic is rerouted to instances in a replacement environment in a blue/green deployment. **deployment\_ready\_option** supports the following:

- **action\_on\_timeout** - (Optional) When to reroute traffic from an original environment to a replacement environment in a blue/green deployment.

- **CONTINUE\_DEPLOYMENT**: Register new instances with the load balancer immediately after the new application revision is installed on the instances in the replacement environment.
- **STOP\_DEPLOYMENT**: Do not register new instances with load balancer unless traffic is rerouted manually. If traffic is not rerouted manually before the end of the specified wait period, the deployment status is changed to Stopped.
- **wait\_time\_in\_minutes** - (Optional) The number of minutes to wait before the status of a blue/green deployment changed to Stopped if rerouting is not started manually. Applies only to the **STOP\_DEPLOYMENT** option for **action\_on\_timeout**.

You can configure how instances will be added to the replacement environment in a blue/green deployment. **green\_fleet\_provisioning\_option** supports the following:

- **action** - (Optional) The method used to add instances to a replacement environment.
  - **DISCOVER\_EXISTING**: Use instances that already exist or will be created manually.
  - **COPY\_AUTO\_SCALING\_GROUP**: Use settings from a specified **Auto Scaling** group to define and create instances in a new Auto Scaling group. *Exactly one Auto Scaling group must be specified* when selecting **COPY\_AUTO\_SCALING\_GROUP**. Use **autoscaling\_groups** to specify the Auto Scaling group.

You can configure how instances in the original environment are terminated when a blue/green deployment is successful. **terminate\_blue\_instances\_on\_deployment\_success** supports the following:

- **action** - (Optional) The action to take on instances in the original environment after a successful blue/green deployment.
  - **TERMINATE**: Instances are terminated after a specified wait time.
  - **KEEP\_ALIVE**: Instances are left running after they are deregistered from the load balancer and removed from the deployment group.
- **termination\_wait\_time\_in\_minutes** - (Optional) The number of minutes to wait after a successful blue/green deployment before terminating instances from the original environment.

## » Attributes Reference

The following attributes are exported:

- **id** - The deployment group's ID.
- **app\_name** - The group's assigned application.

- `deployment_group_name` - The group's name.
- `service_role_arn` - The group's service role ARN.
- `autoscaling_groups` - The autoscaling groups associated with the deployment group.
- `deployment_config_name` - The name of the group's deployment config.

## » `aws__codepipeline`

Provides a CodePipeline.

**NOTE on `aws__codepipeline`:** - the `GITHUB_TOKEN` environment variable must be set if the GitHub provider is specified.

### » Example Usage

```
resource "aws_s3_bucket" "foo" {
  bucket = "test-bucket"
  acl     = "private"
}

resource "aws_iam_role" "foo" {
  name = "test-role"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "codepipeline.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF
}

resource "aws_iam_role_policy" "codepipeline_policy" {
  name = "codepipeline_policy"
  role = "${aws_iam_role.codepipeline_role.id}"
  policy = <<EOF
{
```

```

"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "s3:GetObject",
      "s3:GetObjectVersion",
      "s3:GetBucketVersioning"
    ],
    "Resource": [
      "${aws_s3_bucket.foo.arn}",
      "${aws_s3_bucket.foo.arn}/*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "codebuild:BatchGetBuilds",
      "codebuild:StartBuild"
    ],
    "Resource": "*"
  }
]
}
EOF
}

data "aws_kms_alias" "s3kmskey" {
  name = "alias/myKmsKey"
}

resource "aws_codepipeline" "foo" {
  name      = "tf-test-pipeline"
  role_arn = "${aws_iam_role.foo.arn}"

  artifact_store {
    location = "${aws_s3_bucket.foo.bucket}"
    type     = "S3"
    encryption_key {
      id   = "${data.aws_kms_alias.s3kmskey.arn}"
      type = "KMS"
    }
  }
}

stage {
  name = "Source"

```

```

    action {
        name           = "Source"
        category       = "Source"
        owner          = "ThirdParty"
        provider       = "GitHub"
        version        = "1"
        output_artifacts = ["test"]

        configuration {
            Owner      = "my-organization"
            Repo       = "test"
            Branch     = "master"
        }
    }
}

stage {
    name = "Build"

    action {
        name           = "Build"
        category       = "Build"
        owner          = "AWS"
        provider       = "CodeBuild"
        input_artifacts = ["test"]
        version        = "1"

        configuration {
            ProjectName = "test"
        }
    }
}
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the pipeline.
- **role\_arn** - (Required) A service role Amazon Resource Name (ARN) that grants AWS CodePipeline permission to make calls to AWS services on your behalf.
- **artifact\_store** (Required) An artifact\_store block. Artifact stores are documented below.



- **stage** - (Required) A stage block. Stages are documented below.

An **artifact\_store** block supports the following arguments:

- **location** - (Required) The location where AWS CodePipeline stores artifacts for a pipeline, such as an S3 bucket.
- **type** - (Required) The type of the artifact store, such as Amazon S3
- **encryption\_key** - (Optional) The encryption key block AWS CodePipeline uses to encrypt the data in the artifact store, such as an AWS Key Management Service (AWS KMS) key. If you don't specify a key, AWS CodePipeline uses the default key for Amazon Simple Storage Service (Amazon S3). An **encryption\_key** block is documented below.

A **encryption\_key** block supports the following arguments:

- **id** - (Required) The KMS key ARN or ID
- **type** - (Required) The type of key; currently only KMS is supported

A **stage** block supports the following arguments:

- **name** - (Required) The name of the stage.
- **action** - (Required) The action(s) to include in the stage. Defined as an **action** block below

A **action** block supports the following arguments:

- **category** - (Required) A category defines what kind of action can be taken in the stage, and constrains the provider type for the action. Possible values are **Approval**, **Build**, **Deploy**, **Invoke**, **Source** and **Test**.
- **owner** - (Required) The creator of the action being called. Possible values are **AWS**, **Custom** and **ThirdParty**.
- **name** - (Required) The action declaration's name.
- **provider** - (Required) The provider of the service being called by the action. Valid providers are determined by the action category. For example, an action in the Deploy category type might have a provider of AWS CodeDeploy, which would be specified as CodeDeploy.
- **version** - (Required) A string that identifies the action type.
- **configuration** - (Optional) A Map of the action declaration's configuration. Find out more about configuring action configurations in the Reference Pipeline Structure documentation.
- **input\_artifacts** - (Optional) A list of artifact names to be worked on.
- **output\_artifacts** - (Optional) A list of artifact names to output. Output artifact names must be unique within a pipeline.
- **role\_arn** - (Optional) The ARN of the IAM service role that will perform the declared action. This is assumed through the roleArn for the pipeline.
- **run\_order** - (Optional) The order in which actions are run.

**Note:** The input artifact of an action must exactly match the output artifact declared in a preceding action, but the input artifact does not have to be the

next action in strict sequence from the action that provided the output artifact. Actions in parallel can declare different output artifacts, which are in turn consumed by different following actions.

## » Attributes Reference

The following additional attributes are exported:

- `id` - The codepipeline ID.
- `arn` - The codepipeline ARN.

## » Import

CodePipelines can be imported using the name, e.g.

```
$ terraform import aws_codepipeline.foo example
```

## » `aws_cognito_identity_pool`

Provides an AWS Cognito Identity Pool.

## » Example Usage

```
resource "aws_iam_saml_provider" "default" {
  name                       = "my-saml-provider"
  saml_metadata_document = "${file("saml-metadata.xml")}"
}

resource "aws_cognito_identity_pool" "main" {
  identity_pool_name      = "identity pool"
  allow_unauthenticated_identities = false

  cognito_identity_providers {
    client_id          = "61hlkkfbfb4q5kpp90urffae"
    provider_name      = "cognito-idp.us-east-1.amazonaws.com/us-east-1_Tv0493apJ"
    server_side_token_check = false
  }

  cognito_identity_providers {
    client_id          = "7kodkvfqfb4qfkp39eurffae"
    provider_name      = "cognito-idp.us-east-1.amazonaws.com/eu-west-1_Zr231apJu"
    server_side_token_check = false
  }
}
```

```

}

supported_login_providers {
    "graph.facebook.com" = "7346241598935552"
    "accounts.google.com" = "123456789012.apps.googleusercontent.com"
}

saml_provider_arns          = ["${aws_iam_saml_provider.default.arn}"]
openid_connect_provider_arns = ["arn:aws:iam::123456789012:oidc-provider/foo.example.com"]
}

```

## » Argument Reference

The Cognito Identity Pool argument layout is a structure composed of several sub-resources - these resources are laid out below.

- **identity\_pool\_name** (Required) - The Cognito Identity Pool name.
- **allow\_unauthenticated\_identities** (Required) - Whether the identity pool supports unauthenticated logins or not.
- **developer\_provider\_name** (Optional) - The "domain" by which Cognito will refer to your users. This name acts as a placeholder that allows your backend and the Cognito service to communicate about the developer provider.
- **cognito\_identity\_providers** (Optional) - An array of Amazon Cognito Identity user pools and their client IDs.
- **openid\_connect\_provider\_arns** (Optional) - A list of OpenID Connect provider ARNs.
- **saml\_provider\_arns** (Optional) - An array of Amazon Resource Names (ARNs) of the SAML provider for your identity.
- **supported\_login\_providers** (Optional) - Key-Value pairs mapping provider names to provider app IDs.

## » Cognito Identity Providers

- **client\_id** (Optional) - The client ID for the Amazon Cognito Identity User Pool.
- **provider\_name** (Optional) - The provider name for an Amazon Cognito Identity User Pool.
- **server\_side\_token\_check** (Optional) - Whether server-side token validation is enabled for the identity provider's token or not.

## » Attributes Reference

In addition to the arguments, which are exported, the following attributes are exported:

- `id` - An identity pool ID in the format `REGION:GUID`.

## » Import

Cognito Identity Pool can be imported using the name, e.g.

```
$ terraform import aws_cognito_identity_pool.mypool <identity-pool-id>
```

## » `aws_cognito_identity_pool_roles_attachment`

Provides an AWS Cognito Identity Pool Roles Attachment.

## » Example Usage

```
resource "aws_cognito_identity_pool" "main" {
  identity_pool_name      = "identity pool"
  allow_unauthenticated_identities = false

  supported_login_providers {
    "graph.facebook.com" = "7346241598935555"
  }
}

resource "aws_iam_role" "authenticated" {
  name = "cognito_authenticated"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Federated": "cognito-identity.amazonaws.com"
      },
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
```

```

        "cognito-identity.amazonaws.com:aud": "${aws_cognito_identity_pool.main.id}"
    },
    "ForAnyValue:StringLike": {
        "cognito-identity.amazonaws.com:amr": "authenticated"
    }
}
]
}
EOF
}

resource "aws_iam_role_policy" "authenticated" {
    name = "authenticated_policy"
    role = "${aws_iam_role.authenticated.id}"

    policy = <<EOF
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "mobileanalytics:PutEvents",
                "cognito-sync:*",
                "cognito-identity:*"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
EOF
}

resource "aws_cognito_identity_pool_roles_attachment" "main" {
    identity_pool_id = "${aws_cognito_identity_pool.main.id}"

    role_mapping {
        identity_provider          = "graph.facebook.com"
        ambiguous_role_resolution = "AuthenticatedRole"
        type                      = "Rules"

        mapping_rule {
            claim = "isAdmin"
        }
    }
}

```

```

        match_type = "Equals"
        role_arn    = "${aws_iam_role.authenticated.arn}"
        value       = "paid"
    }
}

roles {
    "authenticated" = "${aws_iam_role.authenticated.arn}"
}
}

```

## » Argument Reference

The Cognito Identity Pool Roles Attachment argument layout is a structure composed of several sub-resources - these resources are laid out below.

- **identity\_pool\_id** (Required) - An identity pool ID in the format REGION:GUID.
- **role\_mapping** (Optional) - A List of Role Mapping.
- **roles** (Required) - The map of roles associated with this pool. For a given role, the key will be either "authenticated" or "unauthenticated" and the value will be the Role ARN.

## » Role Mappings

- **identity\_provider** (Required) - A string identifying the identity provider, for example, "graph.facebook.com" or "cognito-idp-east-1.amazonaws.com/us-east-1\_abcdefghi:app\_client\_id".
- **ambiguous\_role\_resolution** (Optional) - Specifies the action to be taken if either no rules match the claim value for the Rules type, or there is no cognito:preferred\_role claim and there are multiple cognito:roles matches for the Token type. **Required** if you specify Token or Rules as the Type.
- **mapping\_rule** (Optional) - The Rules Configuration to be used for mapping users to roles. You can specify up to 25 rules per identity provider. Rules are evaluated in order. The first one to match specifies the role.
- **type** (Required) - The role mapping type.

## » Rules Configuration

- **claim** (Required) - The claim name that must be present in the token, for example, "isAdmin" or "paid".
- **match\_type** (Required) - The match condition that specifies how closely the claim value in the IdP token must match Value.
- **role\_arn** (Required) - The role ARN.

- **value** (Required) - A brief string that the claim must match, for example, "paid" or "yes".

## » Attributes Reference

In addition to the arguments, which are exported, the following attributes are exported:

- **id** - The identity pool ID.
- **identity\_pool\_id** (Required) - An identity pool ID in the format REGION:GUID.
- **role\_mapping** (Optional) - The List of Role Mapping.
- **roles** (Required) - The map of roles associated with this pool. For a given role, the key will be either "authenticated" or "unauthenticated" and the value will be the Role ARN.

## » `aws_cognito_user_group`

Provides a Cognito User Group resource.

## » Example Usage

```
resource "aws_cognito_user_pool" "main" {
  name = "identity pool"
}

resource "aws_iam_role" "group_role" {
  name = "user-group-role"
  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Federated": "cognito-identity.amazonaws.com"
      },
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
          "cognito-identity.amazonaws.com:aud": "us-east-1:12345678-dead-beef-cafe-12345679"
        }
      }
    }
  ]
}
```

```

        "ForAnyValue:StringLike": {
            "cognito-identity.amazonaws.com:amr": "authenticated"
        }
    }
}
]
}
EOF
}

resource "aws_cognito_user_group" "main" {
    name          = "user-group"
    user_pool_id  = "${aws_cognito_user_pool.main.id}"
    description   = "Managed by Terraform"
    precedence    = 42
    role_arn      = "${aws_iam_role.group_role.arn}"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the user group.
- **user\_pool\_id** - (Required) The user pool ID.
- **description** - (Optional) The description of the user group.
- **precedence** - (Optional) The precedence of the user group.
- **role\_arn** - (Optional) The ARN of the IAM role to be associated with the user group.

## » Import

Cognito User Groups can be imported using the `user_pool_id/name` attributes concatenated, e.g.

```
$ terraform import aws_cognito_user_group.group us-east-1_vG78M4goG/user-group
```

## » `aws_cognito_user_pool`

Provides a Cognito User Pool resource.



## » Example Usage

### » Basic configuration

```
resource "aws_cognito_user_pool" "pool" {  
  name = "mypool"  
}
```

## » Argument Reference

The following arguments are supported:

- **admin\_create\_user\_config** (Optional) - The configuration for Admin-CreateUser requests.
- **alias\_attributes** - (Optional) Attributes supported as an alias for this user pool. Possible values: `phone_number`, `email`, or `preferred_username`. Conflicts with **username\_attributes**.
- **auto\_verified\_attributes** - (Optional) The attributes to be auto-verified. Possible values: `email`, `phone_number`.
- **device\_configuration** (Optional) - The configuration for the user pool's device tracking.
- **email\_configuration** (Optional) - The Email Configuration.
- **name** - (Required) The name of the user pool.
- **email\_verification\_subject** - (Optional) A string representing the email verification subject. **NOTE:** - If **email\_verification\_subject** and **verification\_message\_template.email\_subject** are specified and the values are different, either one is prioritized and updated.
- **email\_verification\_message** - (Optional) A string representing the email verification message. Must contain the `{####}` placeholder. **NOTE:** - If **email\_verification\_message** and **verification\_message\_template.email\_message** are specified and the values are different, either one is prioritized and updated.
- **lambda\_config** (Optional) - A container for the AWS Lambda triggers associated with the user pool.
- **mfa\_configuration** - (Optional, Default: OFF) Set to enable multi-factor authentication. Must be one of the following values (ON, OFF, OPTIONAL)
- **password\_policy** (Optional) - A container for information about the user pool password policy.
- **schema** (Optional) - A container with the schema attributes of a user pool. Maximum of 50 attributes.
- **sms\_authentication\_message** - (Optional) A string representing the SMS authentication message.
- **sms\_configuration** (Optional) - The SMS Configuration.

- `sms_verification_message` - (Optional) A string representing the SMS verification message.
- `tags` - (Optional) A mapping of tags to assign to the User Pool.
- `username_attributes` - (Optional) Specifies whether email addresses or phone numbers can be specified as usernames when a user signs up. Conflicts with `alias_attributes`.
- `verification_message_template` (Optional) - The verification message templates configuration.

#### » Admin Create User Config

- `allow_admin_create_user_only` (Optional) - Set to True if only the administrator is allowed to create user profiles. Set to False if users can sign themselves up via an app.
- `invite_message_template` (Optional) - The invite message template structure.
- `unused_account_validity_days` (Optional) - The user account expiration limit, in days, after which the account is no longer usable.

#### » Invite Message template

- `email_message` (Optional) - The message template for email messages. Must contain `{username}` and `{####}` placeholders, for username and temporary password, respectively.
- `email_subject` (Optional) - The subject line for email messages.
- `sms_message` (Optional) - The message template for SMS messages. Must contain `{username}` and `{####}` placeholders, for username and temporary password, respectively.

#### » Device Configuration

- `challenge_required_on_new_device` (Optional) - Indicates whether a challenge is required on a new device. Only applicable to a new device.
- `device_only_remembered_on_user_prompt` (Optional) - If true, a device is only remembered on user prompt.

#### » Email Configuration

- `reply_to_email_address` (Optional) - The REPLY-TO email address.
- `source_arn` (Optional) - The ARN of the email source.

#### » Lambda Configuration

- `create_auth_challenge` (Optional) - The ARN of the lambda creating an authentication challenge.

- **custom\_message** (Optional) - A custom Message AWS Lambda trigger.
- **define\_auth\_challenge** (Optional) - Defines the authentication challenge.
- **post\_authentication** (Optional) - A post-authentication AWS Lambda trigger.
- **post\_confirmation** (Optional) - A post-confirmation AWS Lambda trigger.
- **pre\_authentication** (Optional) - A pre-authentication AWS Lambda trigger.
- **pre\_sign\_up** (Optional) - A pre-registration AWS Lambda trigger.
- **pre\_token\_generation** (Optional) - Allow to customize identity token claims before token generation.
- **user\_migration** (Optional) - The user migration Lambda config type.
- **verify\_auth\_challenge\_response** (Optional) - Verifies the authentication challenge response.

#### » Password Policy

- **minimum\_length** (Optional) - The minimum length of the password policy that you have set.
- **require\_lowercase** (Optional) - Whether you have required users to use at least one lowercase letter in their password.
- **require\_numbers** (Optional) - Whether you have required users to use at least one number in their password.
- **require\_symbols** (Optional) - Whether you have required users to use at least one symbol in their password.
- **require\_uppercase** (Optional) - Whether you have required users to use at least one uppercase letter in their password.

#### » Schema Attributes

- **attribute\_data\_type** (Required) - The attribute data type. Must be one of Boolean, Number, String, DateTime.
- **developer\_only\_attribute** (Optional) - Specifies whether the attribute type is developer only.
- **mutable** (Optional) - Specifies whether the attribute can be changed once it has been created.
- **name** (Required) - The name of the attribute.
- **number\_attribute\_constraints** (Optional) - Specifies the constraints for an attribute of the number type.
- **required** (Optional) - Specifies whether a user pool attribute is required. If the attribute is required and the user does not provide a value, registration or sign-in will fail.
- **string\_attribute\_constraints** (Optional) - Specifies the constraints for an attribute of the string type.

#### » Number Attribute Constraints

- **max\_value** (Optional) - The maximum value of an attribute that is of the number data type.
- **min\_value** (Optional) - The minimum value of an attribute that is of the number data type.

#### » String Attribute Constraints

- **max\_length** (Optional) - The maximum length of an attribute value of the string type.
- **min\_length** (Optional) - The minimum length of an attribute value of the string type.

#### » SMS Configuration

- **external\_id** (Required) - The external ID used in IAM role trust relationships. For more information about using external IDs, see [How to Use an External ID When Granting Access to Your AWS Resources to a Third Party](#).
- **sns\_caller\_arn** (Required) - The ARN of the Amazon SNS caller. This is usually the IAM role that you've given Cognito permission to assume.

#### » Verification Message Template

- **default\_email\_option** (Optional) - The default email option. Must be either `CONFIRM_WITH_CODE` or `CONFIRM_WITH_LINK`. Defaults to `CONFIRM_WITH_CODE`.
- **email\_message** (Optional) - The email message template. Must contain the `{####}` placeholder. **NOTE:** - If `email_verification_message` and `verification_message_template.email_message` are specified and the values are different, either one is prioritized and updated.
- **email\_message\_by\_link** (Optional) - The email message template for sending a confirmation link to the user, it must contain the `{##VERIFY EMAIL##}` placeholder.
- **email\_subject** (Optional) - The subject line for the email message template. **NOTE:** - If `email_verification_subject` and `verification_message_template.email_subject` are specified and the values are different, either one is prioritized and updated.
- **email\_subject\_by\_link** (Optional) - The subject line for the email message template for sending a confirmation link to the user.
- **sms\_message** (Optional) - The SMS message template. Must contain the `{####}` placeholder.

## » Attribute Reference

The following additional attributes are exported:

- `id` - The id of the user pool.
- `arn` - The ARN of the user pool.
- `creation_date` - The date the user pool was created.
- `last_modified_date` - The date the user pool was last modified.

## » Import

Cognito User Pools can be imported using the `id`, e.g.

```
$ terraform import aws_cognito_user_pool.pool <id>
```

## » `aws_cognito_user_pool_client`

Provides a Cognito User Pool Client resource.

## » Example Usage

### » Create a basic user pool client

```
resource "aws_cognito_user_pool" "pool" {
  name = "pool"
}

resource "aws_cognito_user_pool_client" "client" {
  name = "client"

  user_pool_id = "${aws_cognito_user_pool.pool.id}"
}
```

### » Create a user pool client with no SRP authentication

```
resource "aws_cognito_user_pool" "pool" {
  name = "pool"
}

resource "aws_cognito_user_pool_client" "client" {
  name = "client"
```

```

user_pool_id = "${aws_cognito_user_pool.pool.id}"

generate_secret = true
explicit_auth_flows = ["ADMIN_NO_SRP_AUTH"]
}

```

## » Argument Reference

The following arguments are supported:

- **allowed\_oauth\_flows** - (Optional) List of allowed OAuth flows (code, implicit, client\_credentials).
- **allowed\_oauth\_flows\_user\_pool\_client** - (Optional) Whether the client is allowed to follow the OAuth protocol when interacting with Cognito user pools.
- **allowed\_oauth\_scopes** - (Optional) List of allowed OAuth scopes (phone, email, openid, profile, and aws.cognito.signin.user.admin).
- **callback\_urls** - (Optional) List of allowed callback URLs for the identity providers.
- **default\_redirect\_uri** - (Optional) The default redirect URI. Must be in the list of callback URLs.
- **explicit\_auth\_flows** - (Optional) List of authentication flows (ADMIN\_NO\_SRP\_AUTH, CUSTOM\_AUTH\_FLOW\_ONLY, USER\_PASSWORD\_AUTH).
- **generate\_secret** - (Optional) Should an application secret be generated. AWS JavaScript SDK requires this to be false.
- **logout\_urls** - (Optional) List of allowed logout URLs for the identity providers.
- **name** - (Required) The name of the application client.
- **read\_attributes** - (Optional) List of user pool attributes the application client can read from.
- **refresh\_token\_validity** - (Optional) The time limit in days refresh tokens are valid for.
- **supported\_identity\_providers** - (Optional) List of provider names for the identity providers that are supported on this client.
- **user\_pool\_id** - (Required) The user pool the client belongs to.
- **write\_attributes** - (Optional) List of user pool attributes the application client can write to.

## » Attribute Reference

The following attributes are exported:

- **id** - The id of the user pool client.
- **client\_secret** - The client secret of the user pool client.

## » `aws_cognito_user_pool_domain`

Provides a Cognito User Pool Domain resource.

### » Example Usage

```
resource "aws_cognito_user_pool_domain" "main" {
  domain = "example-domain"
  user_pool_id = "${aws_cognito_user_pool.example.id}"
}

resource "aws_cognito_user_pool" "example" {
  name = "example-pool"
}
```

### » Argument Reference

The following arguments are supported:

- `domain` - (Required) The domain string.
- `user_pool_id` - (Required) The user pool ID.

### » Attribute Reference

The following attributes are exported:

- `aws_account_id` - The AWS account ID for the user pool owner.
- `cloudfront_distribution_arn` - The ARN of the CloudFront distribution.
- `s3_bucket` - The S3 bucket where the static files for this domain are stored.
- `version` - The app version.

## » `aws_config_config_rule`

Provides an AWS Config Rule.

**Note:** Config Rule requires an existing Configuration Recorder to be present. Use of `depends_on` is recommended (as shown below) to avoid race conditions.

## » Example Usage

```
resource "aws_config_config_rule" "r" {
  name = "example"

  source {
    owner          = "AWS"
    source_identifier = "S3_BUCKET_VERSIONING_ENABLED"
  }

  depends_on = ["aws_config_configuration_recorder.foo"]
}

resource "aws_config_configuration_recorder" "foo" {
  name      = "example"
  role_arn = "${aws_iam_role.r.arn}"
}

resource "aws_iam_role" "r" {
  name = "my-awsconfig-role"

  assume_role_policy = <<POLICY
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "config.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
POLICY
}

resource "aws_iam_role_policy" "p" {
  name = "my-awsconfig-policy"
  role = "${aws_iam_role.r.id}"

  policy = <<POLICY
{
  "Version": "2012-10-17",
```



```

    "Statement": [
      {
        "Action": "config:Put*",
        "Effect": "Allow",
        "Resource": "*"
      }
    ]
  }
POLICY
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the rule
- **description** - (Optional) Description of the rule
- **input\_parameters** - (Optional) A string in JSON format that is passed to the AWS Config rule Lambda function.
- **maximum\_execution\_frequency** - (Optional) The maximum frequency with which AWS Config runs evaluations for a rule.
- **scope** - (Optional) Scope defines which resources can trigger an evaluation for the rule as documented below.
- **source** - (Required) Source specifies the rule owner, the rule identifier, and the notifications that cause the function to evaluate your AWS resources as documented below.

### » scope

Defines which resources can trigger an evaluation for the rule. If you do not specify a scope, evaluations are triggered when any resource in the recording group changes.

- **compliance\_resource\_id** - (Optional) The IDs of the only AWS resource that you want to trigger an evaluation for the rule. If you specify a resource ID, you must specify one resource type for **compliance\_resource\_types**.
- **compliance\_resource\_types** - (Optional) A list of resource types of only those AWS resources that you want to trigger an evaluation for the rule. e.g. `AWS::EC2::Instance`. You can only specify one type if you also specify a resource ID for **compliance\_resource\_id**. See relevant part of AWS Docs for available types.
- **tag\_key** - (Optional, Required if **tag\_value** is specified) The tag key that is applied to only those AWS resources that you want you want to trigger an evaluation for the rule.

- **tag\_value** - (Optional) The tag value applied to only those AWS resources that you want to trigger an evaluation for the rule.

#### » **source**

Provides the rule owner (AWS or customer), the rule identifier, and the notifications that cause the function to evaluate your AWS resources.

- **owner** - (Required) Indicates whether AWS or the customer owns and manages the AWS Config rule. The only valid value is **AWS** or **CUSTOM\_LAMBDA**. Keep in mind that Lambda function will require **aws\_lambda\_permission** to allow AWSConfig to execute the function.
- **source\_identifier** - (Required) For AWS Config managed rules, a predefined identifier from a list. For example, **IAM\_PASSWORD\_POLICY** is a managed rule. To reference a managed rule, see Using AWS Managed Config Rules. For custom rules, the identifier is the ARN of the rule's AWS Lambda function, such as **arn:aws:lambda:us-east-1:123456789012:function:custom\_rule\_name**.
- **source\_detail** - (Optional) Provides the source and type of the event that causes AWS Config to evaluate your AWS resources. Only valid if **owner** is **CUSTOM\_LAMBDA**.
  - **event\_source** - (Optional) The source of the event, such as an AWS service, that triggers AWS Config to evaluate your AWS resources. This defaults to **aws.config** and is the only valid value.
  - **maximum\_execution\_frequency** - (Optional) The frequency that you want AWS Config to run evaluations for a rule that is triggered periodically. If specified, requires **message\_type** to be **ScheduledNotification**.
  - **message\_type** - (Optional) The type of notification that triggers AWS Config to run an evaluation for a rule. You can specify the following notification types:
    - \* **ConfigurationItemChangeNotification** - Triggers an evaluation when AWS Config delivers a configuration item as a result of a resource change.
    - \* **OversizedConfigurationItemChangeNotification** - Triggers an evaluation when AWS Config delivers an oversized configuration item. AWS Config may generate this notification type when a resource changes and the notification exceeds the maximum size allowed by Amazon SNS.
    - \* **ScheduledNotification** - Triggers a periodic evaluation at the frequency specified for **maximum\_execution\_frequency**.
    - \* **ConfigurationSnapshotDeliveryCompleted** - Triggers a periodic evaluation when AWS Config delivers a configuration snapshot.

## » Attributes Reference

The following attributes are exported:

- `arn` - The ARN of the config rule
- `rule_id` - The ID of the config rule

## » Import

Config Rule can be imported using the name, e.g.

```
$ terraform import aws_config_config_rule.foo example
```

## » `aws_config_configuration_recorder`

Provides an AWS Config Configuration Recorder. Please note that this resource **does not start** the created recorder automatically.

**Note:** *Starting* the Configuration Recorder requires a delivery channel (while delivery channel creation requires Configuration Recorder). This is why `aws_config_configuration_recorder_status` is a separate resource.

## » Example Usage

```
resource "aws_config_configuration_recorder" "foo" {
  name      = "example"
  role_arn = "${aws_iam_role.r.arn}"
}

resource "aws_iam_role" "r" {
  name = "awsconfig-example"

  assume_role_policy = <<POLICY
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "config.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
```

```

    ]
  }
POLICY
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The name of the recorder. Defaults to **default**. Changing it recreates the resource.
- **role\_arn** - (Required) Amazon Resource Name (ARN) of the IAM role. used to make read or write requests to the delivery channel and to describe the AWS resources associated with the account. See AWS Docs for more details.
- **recording\_group** - (Optional) Recording group - see below.

### » recording\_group

- **all\_supported** - (Optional) Specifies whether AWS Config records configuration changes for every supported type of regional resource (which includes any new type that will become supported in the future). Conflicts with **resource\_types**. Defaults to **true**.
- **include\_global\_resource\_types** - (Optional) Specifies whether AWS Config includes all supported types of *global resources* with the resources that it records. Requires **all\_supported = true**. Conflicts with **resource\_types**.
- **resource\_types** - (Optional) A list that specifies the types of AWS resources for which AWS Config records configuration changes (for example, **AWS::EC2::Instance** or **AWS::CloudTrail::Trail**). See relevant part of AWS Docs for available types.

## » Attributes Reference

The following attributes are exported:

- **id** - Name of the recorder

## » Import

Configuration Recorder can be imported using the name, e.g.

```
$ terraform import aws_config_configuration_recorder.foo example
```

## » `aws_config_configuration_recorder_status`

Manages status (recording / stopped) of an AWS Config Configuration Recorder.

**Note:** Starting Configuration Recorder requires a Delivery Channel to be present. Use of `depends_on` (as shown below) is recommended to avoid race conditions.

### » Example Usage

```
resource "aws_config_configuration_recorder_status" "foo" {
  name          = "${aws_config_configuration_recorder.foo.name}"
  is_enabled    = true
  depends_on    = ["aws_config_delivery_channel.foo"]
}

resource "aws_iam_role_policy_attachment" "a" {
  role          = "${aws_iam_role.r.name}"
  policy_arn    = "arn:aws:iam::aws:policy/service-role/AWSConfigRole"
}

resource "aws_s3_bucket" "b" {
  bucket = "awsconfig-example"
}

resource "aws_config_delivery_channel" "foo" {
  name          = "example"
  s3_bucket_name = "${aws_s3_bucket.b.bucket}"
}

resource "aws_config_configuration_recorder" "foo" {
  name      = "example"
  role_arn  = "${aws_iam_role.r.arn}"
}

resource "aws_iam_role" "r" {
  name = "example-awsconfig"

  assume_role_policy = <<POLICY
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
```

```

        "Principal": {
            "Service": "config.amazonaws.com"
        },
        "Effect": "Allow",
        "Sid": ""
    }
]
}
POLICY
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the recorder
- **is\_enabled** - (Required) Whether the configuration recorder should be enabled or disabled.

## » Import

Configuration Recorder Status can be imported using the name of the Configuration Recorder, e.g.

```
$ terraform import aws_config_configuration_recorder_status.foo example
```

## » aws\_\_config\_\_delivery\_\_channel

Provides an AWS Config Delivery Channel.

**Note:** Delivery Channel requires a Configuration Recorder to be present. Use of `depends_on` (as shown below) is recommended to avoid race conditions.

## » Example Usage

```

resource "aws_config_delivery_channel" "foo" {
    name                = "example"
    s3_bucket_name      = "${aws_s3_bucket.b.bucket}"
    depends_on          = ["aws_config_configuration_recorder.foo"]
}

resource "aws_s3_bucket" "b" {
    bucket              = "example-awsconfig"
}

```

```

    force_destroy = true
  }

  resource "aws_config_configuration_recorder" "foo" {
    name      = "example"
    role_arn = "${aws_iam_role.r.arn}"
  }

  resource "aws_iam_role" "r" {
    name = "awsconfig-example"

    assume_role_policy = <<POLICY
  {
    "Version": "2012-10-17",
    "Statement": [
      {
        "Action": "sts:AssumeRole",
        "Principal": {
          "Service": "config.amazonaws.com"
        },
        "Effect": "Allow",
        "Sid": ""
      }
    ]
  }
  POLICY
  }

  resource "aws_iam_role_policy" "p" {
    name = "awsconfig-example"
    role = "${aws_iam_role.r.id}"

    policy = <<POLICY
  {
    "Version": "2012-10-17",
    "Statement": [
      {
        "Action": [
          "s3:*"
        ],
        "Effect": "Allow",
        "Resource": [
          "${aws_s3_bucket.b.arn}",
          "${aws_s3_bucket.b.arn}/*"
        ]
      }
    ]
  }
  POLICY
  }

```

```

    ]
  }
  POLICY
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The name of the delivery channel. Defaults to **default**. Changing it recreates the resource.
- **s3\_bucket\_name** - (Required) The name of the S3 bucket used to store the configuration history.
- **s3\_key\_prefix** - (Optional) The prefix for the specified S3 bucket.
- **sns\_topic\_arn** - (Optional) The ARN of the SNS topic that AWS Config delivers notifications to.
- **snapshot\_delivery\_properties** - (Optional) Options for how AWS Config delivers configuration snapshots. See below

### » **snapshot\_delivery\_properties**

- **delivery\_frequency** - (Optional) - The frequency with which AWS Config recurrently delivers configuration snapshots. e.g. **One\_Hour** or **Three\_Hours**. Valid values are listed here.

## » Attributes Reference

The following attributes are exported:

- **id** - The name of the delivery channel.

## » Import

Delivery Channel can be imported using the name, e.g.

```
$ terraform import aws_config_delivery_channel.foo example
```

## » **aws\_\_dms\_\_certificate**

Provides a DMS (Data Migration Service) certificate resource. DMS certificates can be created, deleted, and imported.



**Note:** All arguments including the PEM encoded certificate will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Example Usage

```
# Create a new certificate
resource "aws_dms_certificate" "test" {
  certificate_id = "test-dms-certificate-tf"
  certificate_pem = "...
}
```

## » Argument Reference

The following arguments are supported:

- `certificate_id` - (Required) The certificate identifier.
  - Must contain from 1 to 255 alphanumeric characters and hyphens.
- `certificate_pem` - (Optional) The contents of the .pem X.509 certificate file for the certificate. Either `certificate_pem` or `certificate_wallet` must be set.
- `certificate_wallet` - (Optional) The contents of the Oracle Wallet certificate for use with SSL. Either `certificate_pem` or `certificate_wallet` must be set.

## » Attributes Reference

The following attributes are exported:

- `certificate_arn` - The Amazon Resource Name (ARN) for the certificate.

## » Import

Certificates can be imported using the `certificate_arn`, e.g.

```
$ terraform import aws_dms_certificate.test arn:aws:dms:us-west-2:123456789:cert:xxxxxxxxxx
```

## » aws\_\_dms\_\_endpoint

Provides a DMS (Data Migration Service) endpoint resource. DMS endpoints can be created, updated, deleted, and imported.

**Note:** All arguments including the password will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Example Usage

```
# Create a new endpoint
resource "aws_dms_endpoint" "test" {
  certificate_arn      = "arn:aws:acm:us-east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  database_name       = "test"
  endpoint_id         = "test-dms-endpoint-tf"
  endpoint_type       = "source"
  engine_name        = "aurora"
  extra_connection_attributes = ""
  kms_key_arn        = "arn:aws:kms:us-east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
  password            = "test"
  port               = 3306
  server_name         = "test"
  ssl_mode            = "none"

  tags {
    Name = "test"
  }

  username = "test"
}
```

## » Argument Reference

The following arguments are supported:

- **certificate\_arn** - (Optional, Default: empty string) The Amazon Resource Name (ARN) for the certificate.
- **database\_name** - (Optional) The name of the endpoint database.
- **endpoint\_id** - (Required) The database endpoint identifier.
  - Must contain from 1 to 255 alphanumeric characters or hyphens.
  - Must begin with a letter
  - Must contain only ASCII letters, digits, and hyphens
  - Must not end with a hyphen
  - Must not contain two consecutive hyphens
- **endpoint\_type** - (Required) The type of endpoint. Can be one of **source** | **target**.

- **engine\_name** - (Required) The type of engine for the endpoint. Can be one of `mysql` | `oracle` | `postgres` | `mariadb` | `aurora` | `redshift` | `sybase` | `sqlserver` | `dynamodb` | `mongodb`.
- **extra\_connection\_attributes** - (Optional) Additional attributes associated with the connection. For available attributes see Using Extra Connection Attributes with AWS Database Migration Service.
- **kms\_key\_arn** - (Optional) The Amazon Resource Name (ARN) for the KMS key that will be used to encrypt the connection parameters. If you do not specify a value for **kms\_key\_arn**, then AWS DMS will use your default encryption key. AWS KMS creates the default encryption key for your AWS account. Your AWS account has a different default encryption key for each AWS region.
- **password** - (Optional) The password to be used to login to the endpoint database.
- **port** - (Optional) The port used by the endpoint database.
- **server\_name** - (Optional) The host name of the server.
- **ssl\_mode** - (Optional, Default: none) The SSL mode to use for the connection. Can be one of `none` | `require` | `verify-ca` | `verify-full`
- **tags** - (Optional) A mapping of tags to assign to the resource.
- **username** - (Optional) The user name to be used to login to the endpoint database.
- **service\_access\_role** - (Optional) The Amazon Resource Name (ARN) used by the service access IAM role for dynamodb endpoints.
- **mongodb\_settings** - (Optional) Settings for the source MongoDB endpoint. Available settings are **auth\_type** (default: `PASSWORD`), **auth\_mechanism** (default: `DEFAULT`), **nesting\_level** (default: `NONE`), **extract\_doc\_id** (default: `false`), **docs\_to\_investigate** (default: 1000) and **auth\_source** (default: `admin`). For more details, see Using MongoDB as a Source for AWS DMS.

## » Attributes Reference

The following attributes are exported:

- **endpoint\_arn** - The Amazon Resource Name (ARN) for the endpoint.

## » Import

Endpoints can be imported using the **endpoint\_id**, e.g.

```
$ terraform import aws_dms_endpoint.test test-dms-endpoint-tf
```

## » `aws_dms_replication_instance`

Provides a DMS (Data Migration Service) replication instance resource. DMS replication instances can be created, updated, deleted, and imported.

### » Example Usage

```
# Create a new replication instance
resource "aws_dms_replication_instance" "test" {
  allocated_storage           = 20
  apply_immediately           = true
  auto_minor_version_upgrade = true
  availability_zone           = "us-west-2c"
  engine_version              = "1.9.0"
  kms_key_arn                 = "arn:aws:kms:us-east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
  multi_az                    = false
  preferred_maintenance_window = "sun:10:30-sun:14:30"
  publicly_accessible         = true
  replication_instance_class   = "dms.t2.micro"
  replication_instance_id      = "test-dms-replication-instance-tf"
  replication_subnet_group_id = "${aws_dms_replication_subnet_group.test-dms-replication-subnet-group.id}"

  tags {
    Name = "test"
  }

  vpc_security_group_ids = [
    "sg-12345678",
  ]
}
```

### » Argument Reference

The following arguments are supported:

- `allocated_storage` - (Optional, Default: 50, Min: 5, Max: 6144) The amount of storage (in gigabytes) to be initially allocated for the replication instance.
- `apply_immediately` - (Optional, Default: false) Indicates whether the changes should be applied immediately or during the next maintenance

window. Only used when updating an existing resource.

- **auto\_minor\_version\_upgrade** - (Optional, Default: false) Indicates that minor engine upgrades will be applied automatically to the replication instance during the maintenance window.
- **availability\_zone** - (Optional) The EC2 Availability Zone that the replication instance will be created in.
- **engine\_version** - (Optional) The engine version number of the replication instance.
- **kms\_key\_arn** - (Optional) The Amazon Resource Name (ARN) for the KMS key that will be used to encrypt the connection parameters. If you do not specify a value for **kms\_key\_arn**, then AWS DMS will use your default encryption key. AWS KMS creates the default encryption key for your AWS account. Your AWS account has a different default encryption key for each AWS region.
- **multi\_az** - (Optional) Specifies if the replication instance is a multi-az deployment. You cannot set the **availability\_zone** parameter if the **multi\_az** parameter is set to **true**.
- **preferred\_maintenance\_window** - (Optional) The weekly time range during which system maintenance can occur, in Universal Coordinated Time (UTC).
  - Default: A 30-minute window selected at random from an 8-hour block of time per region, occurring on a random day of the week.
  - Format: **ddd:hh24:mi-ddd:hh24:mi**
  - Valid Days: **mon, tue, wed, thu, fri, sat, sun**
  - Constraints: Minimum 30-minute window.
- **publicly\_accessible** - (Optional, Default: false) Specifies the accessibility options for the replication instance. A value of **true** represents an instance with a public IP address. A value of **false** represents an instance with a private IP address.
- **replication\_instance\_class** - (Required) The compute and memory capacity of the replication instance as specified by the replication instance class. Can be one of **dms.t2.micro** | **dms.t2.small** | **dms.t2.medium** | **dms.t2.large** | **dms.c4.large** | **dms.c4.xlarge** | **dms.c4.2xlarge** | **dms.c4.4xlarge**
- **replication\_instance\_id** - (Required) The replication instance identifier. This parameter is stored as a lowercase string.
  - Must contain from 1 to 63 alphanumeric characters or hyphens.
  - First character must be a letter.
  - Cannot end with a hyphen
  - Cannot contain two consecutive hyphens.

- **replication\_subnet\_group\_id** - (Optional) A subnet group to associate with the replication instance.
- **tags** - (Optional) A mapping of tags to assign to the resource.
- **vpc\_security\_group\_ids** - (Optional) A list of VPC security group IDs to be used with the replication instance. The VPC security groups must work with the VPC containing the replication instance.

## » Attributes Reference

The following attributes are exported:

- **replication\_instance\_arn** - The Amazon Resource Name (ARN) of the replication instance.
- **replication\_instance\_private\_ips** - A list of the private IP addresses of the replication instance.
- **replication\_instance\_public\_ips** - A list of the public IP addresses of the replication instance.

## » Timeouts

`aws_dms_replication_instance` provides the following Timeouts configuration options:

- **create** - (Default 30 minutes) Used for Creating Instances
- **update** - (Default 30 minutes) Used for Database modifications
- **delete** - (Default 30 minutes) Used for destroying databases.

## » Import

Replication instances can be imported using the `replication_instance_id`, e.g.

```
$ terraform import aws_dms_replication_instance.test test-dms-replication-instance-tf
```

## » `aws_dms_replication_subnet_group`

Provides a DMS (Data Migration Service) replication subnet group resource. DMS replication subnet groups can be created, updated, deleted, and imported.

## » Example Usage

```
# Create a new replication subnet group
resource "aws_dms_replication_subnet_group" "test" {
  replication_subnet_group_description = "Test replication subnet group"
  replication_subnet_group_id         = "test-dms-replication-subnet-group-tf"

  subnet_ids = [
    "subnet-12345678",
  ]
}
```

## » Argument Reference

The following arguments are supported:

- `replication_subnet_group_description` - (Required) The description for the subnet group.
- `replication_subnet_group_id` - (Required) The name for the replication subnet group. This value is stored as a lowercase string.
  - Must contain no more than 255 alphanumeric characters, periods, spaces, underscores, or hyphens.
  - Must not be "default".
- `subnet_ids` - (Required) A list of the EC2 subnet IDs for the subnet group.

## » Attributes Reference

The following attributes are exported:

- `vpc_id` - The ID of the VPC the subnet group is in.

## » Import

Replication subnet groups can be imported using the `replication_subnet_group_id`, e.g.

```
$ terraform import aws_dms_replication_subnet_group.test test-dms-replication-subnet-group-tf
```

## » `aws_dms_replication_task`

Provides a DMS (Data Migration Service) replication task resource. DMS replication tasks can be created, updated, deleted, and imported.

### » Example Usage

```
# Create a new replication task
resource "aws_dms_replication_task" "test" {
  cdc_start_time          = 1484346880
  migration_type           = "full-load"
  replication_instance_arn = "${aws_dms_replication_instance.test-dms-replication-instance.arn}"
  replication_task_id      = "test-dms-replication-task-tf"
  replication_task_settings = "...
  source_endpoint_arn      = "${aws_dms_endpoint.test-dms-source-endpoint-tf.endpoint_arn}"
  table_mappings            = "{\"rules\": [{\"rule-type\": \"selection\", \"rule-id\": \"1\", \"rule-name\": \"select-all-tables\"}]}"

  tags {
    Name = "test"
  }
}

target_endpoint_arn = "${aws_dms_endpoint.test-dms-target-endpoint-tf.endpoint_arn}"
}
```

### » Argument Reference

The following arguments are supported:

- `cdc_start_time` - (Optional) The Unix timestamp integer for the start of the Change Data Capture (CDC) operation.
- `migration_type` - (Required) The migration type. Can be one of `full-load` | `cdc` | `full-load-and-cdc`.
- `replication_instance_arn` - (Required) The Amazon Resource Name (ARN) of the replication instance.
- `replication_task_id` - (Required) The replication task identifier.
  - Must contain from 1 to 255 alphanumeric characters or hyphens.
  - First character must be a letter.
  - Cannot end with a hyphen.
  - Cannot contain two consecutive hyphens.
- `replication_task_settings` - (Optional) An escaped JSON string that contains the task settings. For a complete list of task settings, see Task Settings for AWS Database Migration Service Tasks.



- **source\_endpoint\_arn** - (Required) The Amazon Resource Name (ARN) string that uniquely identifies the source endpoint.
- **table\_mappings** - (Required) An escaped JSON string that contains the table mappings. For information on table mapping see [Using Table Mapping with an AWS Database Migration Service Task to Select and Filter Data](#)
- **tags** - (Optional) A mapping of tags to assign to the resource.
- **target\_endpoint\_arn** - (Required) The Amazon Resource Name (ARN) string that uniquely identifies the target endpoint.

## » Attributes Reference

The following attributes are exported:

- **replication\_task\_arn** - The Amazon Resource Name (ARN) for the replication task.

## » Import

Replication tasks can be imported using the **replication\_task\_id**, e.g.

```
$ terraform import aws_dms_replication_task.test test-dms-replication-task-tf
```

## » `aws__devicefarm__project`

Provides a resource to manage AWS Device Farm Projects. Please keep in mind that this feature is only supported on the "us-west-2" region. This resource will error if you try to create a project in another region.

For more information about Device Farm Projects, see the [AWS Documentation on Device Farm Projects](#).

## » Basic Example Usage

```
resource "aws_devicefarm_project" "awesome_devices" {
  name = "my-device-farm"
}
```

## » Argument Reference

- **name** - (Required) The name of the project

## » Attributes Reference

The following attributes are exported:

- `arn` - The Amazon Resource Name of this project

## » `aws_directory_service_directory`

Provides a Simple or Managed Microsoft directory in AWS Directory Service.

**Note:** All arguments including the password and customer username will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Example Usage

### » SimpleAD

```
resource "aws_directory_service_directory" "bar" {
  name      = "corp.notexample.com"
  password  = "SuperSecretPassw0rd"
  size      = "Small"

  vpc_settings {
    vpc_id      = "${aws_vpc.main.id}"
    subnet_ids = ["${aws_subnet.foo.id}", "${aws_subnet.bar.id}"]
  }

  tags {
    Project = "foo"
  }
}

resource "aws_vpc" "main" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_subnet" "foo" {
  vpc_id            = "${aws_vpc.main.id}"
  availability_zone = "us-west-2a"
  cidr_block        = "10.0.1.0/24"
}

resource "aws_subnet" "bar" {
  vpc_id = "${aws_vpc.main.id}"
}
```

```

    availability_zone = "us-west-2b"
    cidr_block        = "10.0.2.0/24"
}

```

## » Microsoft Active Directory (MicrosoftAD)

```

resource "aws_directory_service_directory" "bar" {
  name      = "corp.notexample.com"
  password  = "SuperSecretPassw0rd"
  edition   = "Standard"
  type      = "MicrosoftAD"

  vpc_settings {
    vpc_id      = "${aws_vpc.main.id}"
    subnet_ids = ["${aws_subnet.foo.id}", "${aws_subnet.bar.id}"]
  }

  tags {
    Project = "foo"
  }
}

resource "aws_vpc" "main" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_subnet" "foo" {
  vpc_id            = "${aws_vpc.main.id}"
  availability_zone = "us-west-2a"
  cidr_block        = "10.0.1.0/24"
}

resource "aws_subnet" "bar" {
  vpc_id            = "${aws_vpc.main.id}"
  availability_zone = "us-west-2b"
  cidr_block        = "10.0.2.0/24"
}

```

## » Microsoft Active Directory Connector (ADConnector)

```

resource "aws_directory_service_directory" "connector" {
  name      = "corp.notexample.com"
  password  = "SuperSecretPassw0rd"
  size      = "Small"
}

```

```

type      = "ADConnector"

connect_settings {
  customer_dns_ips   = ["A.B.C.D"]
  customer_username  = "Administrator"
  subnet_ids        = ["${aws_subnet.foo.id}", "${aws_subnet.bar.id}"]
  vpc_id             = "${aws_vpc.main.id}"
}
}

resource "aws_vpc" "main" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_subnet" "foo" {
  vpc_id            = "${aws_vpc.main.id}"
  availability_zone = "us-west-2a"
  cidr_block        = "10.0.1.0/24"
}

resource "aws_subnet" "bar" {
  vpc_id            = "${aws_vpc.main.id}"
  availability_zone = "us-west-2b"
  cidr_block        = "10.0.2.0/24"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The fully qualified name for the directory, such as `corp.example.com`
- **password** - (Required) The password for the directory administrator or connector user.
- **size** - (Required for `SimpleAD` and `ADConnector`) The size of the directory (`Small` or `Large` are accepted values).
- **vpc\_settings** - (Required for `SimpleAD` and `MicrosoftAD`) VPC related information about the directory. Fields documented below.
- **connect\_settings** - (Required for `ADConnector`) Connector related information about the directory. Fields documented below.
- **alias** - (Optional) The alias for the directory (must be unique amongst all aliases in AWS). Required for `enable_sso`.
- **description** - (Optional) A textual description for the directory.
- **short\_name** - (Optional) The short name of the directory, such as `CORP`.
- **enable\_sso** - (Optional) Whether to enable single-sign on for the directory.

Requires `alias`. Defaults to `false`.

- `type` (Optional) - The directory type (`SimpleAD`, `ADConnector` or `MicrosoftAD` are accepted values). Defaults to `SimpleAD`.
- `edition` - (Optional) The MicrosoftAD edition (`Standard` or `Enterprise`). Defaults to `Enterprise` (applies to MicrosoftAD type only).
- `tags` - (Optional) A mapping of tags to assign to the resource.

`vpc_settings` supports the following:

- `subnet_ids` - (Required) The identifiers of the subnets for the directory servers (2 subnets in 2 different AZs).
- `vpc_id` - (Required) The identifier of the VPC that the directory is in.

`connect_settings` supports the following:

- `customer_username` - (Required) The username corresponding to the password provided.
- `customer_dns_ips` - (Required) The DNS IP addresses of the domain to connect to.
- `subnet_ids` - (Required) The identifiers of the subnets for the directory servers (2 subnets in 2 different AZs).
- `vpc_id` - (Required) The identifier of the VPC that the directory is in.

## » Attributes Reference

The following attributes are exported:

- `id` - The directory identifier.
- `access_url` - The access URL for the directory, such as `http://alias.awsapps.com`.
- `dns_ip_addresses` - A list of IP addresses of the DNS servers for the directory or connector.
- `security_group_id` - The ID of the security group created by the directory (`SimpleAD` or `MicrosoftAD` only).

## » Import

DirectoryService directories can be imported using the directory `id`, e.g.

```
$ terraform import aws_directory_service_directory.sample d-926724cf57
```

## » `aws_directory_service_conditional_forwarder`

Provides a conditional forwarder for managed Microsoft AD in AWS Directory Service.

## » Example Usage

```
resource "aws_directory_service_conditional_forwarder" "example" {
  directory_id = "${aws_directory_service_directory.ad.id}"
  remote_domain_name = "example.com"

  dns_ips = [
    "8.8.8.8",
    "8.8.4.4",
  ]
}
```

## » Argument Reference

The following arguments are supported:

- `directory_id` - (Required) The id of directory.
- `dns_ips` - (Required) A list of forwarder IP addresses.
- `remote_domain_name` - (Required) The fully qualified domain name of the remote domain for which forwarders will be used.

## » Import

Conditional forwarders can be imported using the directory id and remote\_domain\_name, e.g.

```
$ terraform import aws_directory_service_conditional_forwarder.example d-1234567890:example
```

## » aws\_\_dx\_\_connection

Provides a Connection of Direct Connect.

## » Example Usage

```
resource "aws_dx_connection" "hoge" {
  name = "tf-dx-connection"
  bandwidth = "1Gbps"
  location = "EqDC2"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the connection.
- **bandwidth** - (Required) The bandwidth of the connection. Available values: 1Gbps, 10Gbps. Case sensitive.
- **location** - (Required) The AWS Direct Connect location where the connection is located. See `DescribeLocations` for the list of AWS Direct Connect locations. Use `locationCode`.
- **tags** - (Optional) A mapping of tags to assign to the resource.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the connection.
- **arn** - The ARN of the connection.

## » Import

Direct Connect connections can be imported using the `connection id`, e.g.

```
$ terraform import aws_dx_connection.test_connection dxcon-ffre0ec3
```

## » `aws_dx_connection_association`

Associates a Direct Connect Connection with a LAG.

## » Example Usage

```
resource "aws_dx_connection" "example" {
  name = "example"
  bandwidth = "1Gbps"
  location = "EqSe2"
}

resource "aws_dx_lag" "example" {
  name = "example"
  connections_bandwidth = "1Gbps"
  location = "EqSe2"
  number_of_connections = 1
}
```

```
resource "aws_dx_connection_association" "example" {
  connection_id = "${aws_dx_connection.example.id}"
  lag_id = "${aws_dx_lag.example.id}"
}
```

## » Argument Reference

The following arguments are supported:

- `connection_id` - (Required) The ID of the connection.
- `lag_id` - (Required) The ID of the LAG with which to associate the connection.

## » `aws_dx_lag`

Provides a Direct Connect LAG.

## » Example Usage

```
resource "aws_dx_lag" "hoge" {
  name = "tf-dx-lag"
  connections_bandwidth = "1Gbps"
  location = "EqDC2"
  number_of_connections = 2
  force_destroy = true
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the LAG.
- `connections_bandwidth` - (Required) The bandwidth of the individual physical connections bundled by the LAG. Available values: 1Gbps, 10Gbps. Case sensitive.
- `location` - (Required) The AWS Direct Connect location in which the LAG should be allocated. See `DescribeLocations` for the list of AWS Direct Connect locations. Use `locationCode`.
- `number_of_connections` - (**Deprecated**) The number of physical connections initially provisioned and bundled by the LAG. Use `aws_dx_connection` and `aws_dx_connection_association` resources



instead. Default connections will be removed as part of LAG creation automatically in future versions.

- **force\_destroy** - (Optional, Default:false) A boolean that indicates all connections associated with the LAG should be deleted so that the LAG can be destroyed without error. These objects are *not* recoverable.
- **tags** - (Optional) A mapping of tags to assign to the resource.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the LAG.
- **arn** - The ARN of the LAG.

## » Import

Direct Connect LAGs can be imported using the `lag id`, e.g.

```
$ terraform import aws_dx_lag.test_lag dxlag-fgnsp5rq
```

## » aws\_\_dynamodb\_\_global\_\_table

Provides a resource to manage a DynamoDB Global Table. These are layered on top of existing DynamoDB Tables.

Note: There are many restrictions before you can properly create DynamoDB Global Tables in multiple regions. See the AWS DynamoDB Global Table Requirements for more information.

## » Example Usage

```
provider "aws" {
  alias   = "us-east-1"
  region = "us-east-1"
}

provider "aws" {
  alias   = "us-west-2"
  region = "us-west-2"
}

resource "aws_dynamodb_table" "us-east-1" {
  provider = "aws.us-east-1"
```

```

    hash_key      = "myAttribute"
    name          = "myTable"
    stream_enabled = true
    stream_view_type = "NEW_AND_OLD_IMAGES"
    read_capacity  = 1
    write_capacity = 1

    attribute {
      name = "myAttribute"
      type = "S"
    }
  }

resource "aws_dynamodb_table" "us-west-2" {
  provider = "aws.us-west-2"

  hash_key      = "myAttribute"
  name          = "myTable"
  stream_enabled = true
  stream_view_type = "NEW_AND_OLD_IMAGES"
  read_capacity  = 1
  write_capacity = 1

  attribute {
    name = "myAttribute"
    type = "S"
  }
}

resource "aws_dynamodb_global_table" "myTable" {
  depends_on = ["aws_dynamodb_table.us-east-1", "aws_dynamodb_table.us-west-2"]
  provider   = "aws.us-east-1"

  name = "myTable"

  replica {
    region_name = "us-east-1"
  }

  replica {
    region_name = "us-west-2"
  }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the global table. Must match underlying DynamoDB Table names in all regions.
- **replica** - (Required) Underlying DynamoDB Table. At least 1 replica must be defined. See below.

## » Nested Fields

### » **replica**

- **region\_name** - (Required) AWS region name of replica DynamoDB Table.  
e.g. `us-east-1`

## » Attributes Reference

The following additional attributes are exported:

- **id** - The name of the DynamoDB Global Table
- **arn** - The ARN of the DynamoDB Global Table

## » Import

DynamoDB Global Tables can be imported using the global table name, e.g.

```
$ terraform import aws_dynamodb_global_table.MyTable MyTable
```

## » **aws\_\_dynamodb\_\_table**

Provides a DynamoDB table resource

**Note:** It is recommended to use `lifecycle ignore_changes` for `read_capacity` and/or `write_capacity` if there's autoscaling policy attached to the table.

## » Example Usage

The following dynamodb table description models the table and GSI shown in the AWS SDK example documentation

```

resource "aws_dynamodb_table" "basic-dynamodb-table" {
  name           = "GameScores"
  read_capacity  = 20
  write_capacity = 20
  hash_key       = "UserId"
  range_key      = "GameTitle"

  attribute {
    name = "UserId"
    type = "S"
  }

  attribute {
    name = "GameTitle"
    type = "S"
  }

  attribute {
    name = "TopScore"
    type = "N"
  }

  ttl {
    attribute_name = "TimeToExist"
    enabled        = false
  }

  global_secondary_index {
    name           = "GameTitleIndex"
    hash_key       = "GameTitle"
    range_key      = "TopScore"
    write_capacity = 10
    read_capacity  = 10
    projection_type = "INCLUDE"
    non_key_attributes = ["UserId"]
  }

  tags {
    Name           = "dynamodb-table-1"
    Environment    = "production"
  }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the table, this needs to be unique within a region.
- **hash\_key** - (Required, Forces new resource) The attribute to use as the hash key (the attribute must also be defined as an attribute record)
- **range\_key** - (Optional, Forces new resource) The attribute to use as the range key (must also be defined)
- **write\_capacity** - (Required) The number of write units for this table
- **read\_capacity** - (Required) The number of read units for this table
- **attribute** - (Required) Define an attribute, has two properties:
  - **name** - The name of the attribute
  - **type** - One of: S, N, or B for (S)tring, (N)umber or (B)inary data
- **ttl** - (Optional) Defines ttl, has two properties, and can only be specified once:
  - **enabled** - (Required) Indicates whether ttl is enabled (true) or disabled (false).
  - **attribute\_name** - (Required) The name of the table attribute to store the TTL timestamp in.
- **local\_secondary\_index** - (Optional, Forces new resource) Describe an LSI on the table; these can only be allocated *at creation* so you cannot change this definition after you have created the resource.
- **global\_secondary\_index** - (Optional) Describe a GSO for the table; subject to the normal limits on the number of GSIs, projected attributes, etc.
- **stream\_enabled** - (Optional) Indicates whether Streams are to be enabled (true) or disabled (false).
- **stream\_view\_type** - (Optional) When an item in the table is modified, StreamViewType determines what information is written to the table's stream. Valid values are KEYS\_ONLY, NEW\_IMAGE, OLD\_IMAGE, NEW\_AND\_OLD\_IMAGES.
- **server\_side\_encryption** - (Optional) Encrypt at rest options.
- **tags** - (Optional) A map of tags to populate on the created table.
- **point\_in\_time\_recovery** - (Optional) Point-in-time recovery options.

## » Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- **create** - (Defaults to 10 mins) Used when creating the table
- **update** - (Defaults to 10 mins) Used when updating the table
- **delete** - (Defaults to 10 mins) Used when deleting the table

» **Nested fields**

» **local\_secondary\_index**

- **name** - (Required) The name of the index
- **range\_key** - (Required) The name of the range key; must be defined
- **projection\_type** - (Required) One of **ALL**, **INCLUDE** or **KEYS\_ONLY** where **ALL** projects every attribute into the index, **KEYS\_ONLY** projects just the hash and range key into the index, and **INCLUDE** projects only the keys specified in the *non\_key\_attributes* parameter.
- **non\_key\_attributes** - (Optional) Only required with **INCLUDE** as a projection type; a list of attributes to project into the index. These do not need to be defined as attributes on the table.

» **global\_secondary\_index**

- **name** - (Required) The name of the index
- **write\_capacity** - (Required) The number of write units for this index
- **read\_capacity** - (Required) The number of read units for this index
- **hash\_key** - (Required) The name of the hash key in the index; must be defined as an attribute in the resource.
- **range\_key** - (Optional) The name of the range key; must be defined
- **projection\_type** - (Required) One of **ALL**, **INCLUDE** or **KEYS\_ONLY** where **ALL** projects every attribute into the index, **KEYS\_ONLY** projects just the hash and range key into the index, and **INCLUDE** projects only the keys specified in the *non\_key\_attributes* parameter.
- **non\_key\_attributes** - (Optional) Only required with **INCLUDE** as a projection type; a list of attributes to project into the index. These do not need to be defined as attributes on the table.

» **server\_side\_encryption**

- **enabled** - (Required) Whether to enable encryption at rest. If the **server\_side\_encryption** block is not provided then this defaults to **false**.

» **point\_in\_time\_recovery**

- **enabled** - (Required) Whether to enable point-in-time recovery - note that it can take up to 10 minutes to enable for new tables. If the **point\_in\_time\_recovery** block is not provided then this defaults to **false**.

## » A note about attributes

Only define attributes on the table object that are going to be used as:

- Table hash key or range key
- LSI or GSI hash key or range key

The DynamoDB API expects attribute structure (name and type) to be passed along when creating or updating GSI/LSIs or creating the initial table. In these cases it expects the Hash / Range keys to be provided; because these get re-used in numerous places (i.e the table's range key could be a part of one or more GSIs), they are stored on the table object to prevent duplication and increase consistency. If you add attributes here that are not used in these scenarios it can cause an infinite loop in planning.

## » Attributes Reference

The following attributes are exported:

- **arn** - The arn of the table
- **id** - The name of the table
- **stream\_arn** - The ARN of the Table Stream. Only available when **stream\_enabled = true**
- **stream\_label** - A timestamp, in ISO 8601 format, for this stream. Note that this timestamp is not a unique identifier for the stream on its own. However, the combination of AWS customer ID, table name and this field is guaranteed to be unique. It can be used for creating CloudWatch Alarms. Only available when **stream\_enabled = true**

## » Import

DynamoDB tables can be imported using the **name**, e.g.

```
$ terraform import aws_dynamodb_table.basic-dynamodb-table GameScores
```

## » `aws__dynamodb__table__item`

Provides a DynamoDB table item resource

**Note:** This resource is not meant to be used for managing large amounts of data in your table, it is not designed to scale. You should perform **regular backups** of all data in the table, see AWS docs for more.

## » Example Usage

```
resource "aws_dynamodb_table_item" "example" {
  table_name = "${aws_dynamodb_table.example.name}"
  hash_key = "${aws_dynamodb_table.example.hash_key}"
  item = <<ITEM
{
  "exampleHashKey": {"S": "something"},
  "one": {"N": "11111"},
  "two": {"N": "22222"},
  "three": {"N": "33333"},
  "four": {"N": "44444"}
}
ITEM
}
```

```
resource "aws_dynamodb_table" "example" {
  name = "example-name"
  read_capacity = 10
  write_capacity = 10
  hash_key = "exampleHashKey"

  attribute {
    name = "exampleHashKey"
    type = "S"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **table\_name** - (Required) The name of the table to contain the item.
- **hash\_key** - (Required) Hash key to use for lookups and identification of the item
- **range\_key** - (Optional) Range key to use for lookups and identification of the item. Required if there is range key defined in the table.
- **item** - (Required) JSON representation of a map of attribute name/value pairs, one for each attribute. Only the primary key attributes are required; you can optionally provide other attribute name-value pairs for the item.

## » Attributes Reference

All of the arguments above are exported as attributes.



## » Import

DynamoDB table items cannot be imported.

## » `aws__dax__cluster`

Provides a DAX Cluster resource.

## » Example Usage

```
resource "aws_dax_cluster" "bar" {
  cluster_id      = "cluster-example"
  iam_role_arn    = "${data.aws_iam_role.example.arn}"
  node_type       = "dax.r3.large"
  replication_factor = 1
}
```

## » Argument Reference

The following arguments are supported:

- `cluster_name` – (Required) Group identifier. DAX converts this name to lowercase
- `iam_role_arn` - (Required) A valid Amazon Resource Name (ARN) that identifies an IAM role. At runtime, DAX will assume this role and use the role's permissions to access DynamoDB on your behalf
- `node_type` – (Required) The compute and memory capacity of the nodes. See Nodes for supported node types
- `replication_factor` – (Required) The number of nodes in the DAX cluster. A replication factor of 1 will create a single-node cluster, without any read replicas
- `availability_zones` - (Optional) List of Availability Zones in which the nodes will be created
- `description` – (Optional) Description for the cluster
- `notification_topic_arn` – (Optional) An Amazon Resource Name (ARN) of an SNS topic to send DAX notifications to. Example: `arn:aws:sns:us-east-1:012345678999:my_sns_topic`
- `parameter_group_name` – (Optional) Name of the parameter group to associate with this DAX cluster

- **maintenance\_window** – (Optional) Specifies the weekly time range for when maintenance on the cluster is performed. The format is `ddd:hh24:mi-ddd:hh24:mi` (24H Clock UTC). The minimum maintenance window is a 60 minute period. Example: `sun:05:00-sun:09:00`
- **security\_group\_ids** – (Optional) One or more VPC security groups associated with the cluster
- **subnet\_group\_name** – (Optional) Name of the subnet group to be used for the cluster
- **tags** - (Optional) A mapping of tags to assign to the resource

## » Attributes Reference

The following additional attributes are exported:

- **arn** - The ARN of the DAX cluster
- **nodes** - List of node objects including `id`, `address`, `port` and `availability_zone`. Referenceable e.g. as `${aws_dax_cluster.test.nodes.0.address}`
- **configuration\_endpoint** - The configuration endpoint for this DAX cluster, consisting of a DNS name and a port number
- **cluster\_address** - The DNS name of the DAX cluster without the port appended
- **port** - The port used by the configuration endpoint

## » Timeouts

`aws_dax_cluster` provides the following Timeouts configuration options:

- **create** - (Default 45 minutes) Used for creating a DAX cluster
- **update** - (Default 45 minutes) Used for cluster modifications
- **delete** - (Default 90 minutes) Used for destroying a DAX cluster

## » Import

DAX Clusters can be imported using the `cluster_id`, e.g.

```
$ terraform import aws_dax_cluster.my_cluster my_cluster
```

## » `aws__dax__parameter__group`

Provides a DAX Parameter Group resource.

## » Example Usage

```
resource "aws_dax_parameter_group" "example" {
  name = "example"
  parameters {
    name = "query-ttl-millis"
    value = "100000"
  }
  parameters {
    name = "record-ttl-millis"
    value = "100000"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** – (Required) The name of the parameter group.
- **description** - (Optional, ForceNew) A description of the parameter group.
- **parameters** – (Optional) The parameters of the parameter group.

## » parameters

**parameters** supports the following:

- **name** - (Required) The name of the parameter.
- **value** - (Required) The value for the parameter.

## » Attributes Reference

The following additional attributes are exported:

- **id** - The name of the parameter group.

## » Import

DAX Parameter Group can be imported using the **name**, e.g.

```
$ terraform import aws_dax_parameter_group.example my_dax_pg
```

## » **aws\_\_dax\_\_subnet\_\_group**

Provides a DAX Subnet Group resource.

### » **Example Usage**

```
resource "aws_dax_subnet_group" "example" {  
  name = "example"  
  subnet_ids = ["${aws_subnet.example1.id}", "${aws_subnet.example2.id}"]  
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** – (Required) The name of the subnet group.
- **description** – (Optional) A description of the subnet group.
- **subnet\_ids** – (Required) A list of VPC subnet IDs for the subnet group.

### » **Attributes Reference**

The following additional attributes are exported:

- **id** – The name of the subnet group.
- **vpc\_id** – VPC ID of the subnet group.

### » **Import**

DAX Subnet Group can be imported using the **name**, e.g.

```
$ terraform import aws_dax_subnet_group.example my_dax_sg
```

## » **aws\_\_lb**

Provides a Load Balancer resource.

**Note:** `aws_alb` is known as `aws_lb`. The functionality is identical.

## » Example Usage

### » Application Load Balancer

```
resource "aws_lb" "test" {
  name                = "test-lb-tf"
  internal            = false
  load_balancer_type = "application"
  security_groups     = ["${aws_security_group.lb_sg.id}"]
  subnets            = ["${aws_subnet.public.*.id}"]

  enable_deletion_protection = true

  access_logs {
    bucket = "${aws_s3_bucket.lb_logs.bucket}"
    prefix = "test-lb"
    enabled = true
  }

  tags {
    Environment = "production"
  }
}
```

### » Network Load Balancer

```
resource "aws_lb" "test" {
  name                = "test-lb-tf"
  internal            = false
  load_balancer_type = "network"
  subnets            = ["${aws_subnet.public.*.id}"]

  enable_deletion_protection = true

  tags {
    Environment = "production"
  }
}
```

### » Specifying Elastic IPs

```
resource "aws_lb" "example" {
  name                = "example"
  load_balancer_type = "network"
```

```

subnet_mapping {
  subnet_id      = "${aws_subnet.example1.id}"
  allocation_id = "${aws_eip.example1.id}"
}

subnet_mapping {
  subnet_id      = "${aws_subnet.example2.id}"
  allocation_id = "${aws_eip.example2.id}"
}
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The name of the LB. This name must be unique within your AWS account, can have a maximum of 32 characters, must contain only alphanumeric characters or hyphens, and must not begin or end with a hyphen. If not specified, Terraform will autogenerate a name beginning with **tf-lb**.
- **name\_prefix** - (Optional) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **internal** - (Optional) If true, the LB will be internal.
- **load\_balancer\_type** - (Optional) The type of load balancer to create. Possible values are **application** or **network**. The default value is **application**.
- **security\_groups** - (Optional) A list of security group IDs to assign to the LB. Only valid for Load Balancers of type **application**.
- **access\_logs** - (Optional) An Access Logs block. Access Logs documented below. Only valid for Load Balancers of type **application**.
- **subnets** - (Optional) A list of subnet IDs to attach to the LB. Subnets cannot be updated for Load Balancers of type **network**. Changing this value for load balancers of type **network** will force a recreation of the resource.
- **subnet\_mapping** - (Optional) A subnet mapping block as documented below.
- **idle\_timeout** - (Optional) The time in seconds that the connection is allowed to be idle. Only valid for Load Balancers of type **application**. Default: 60.
- **enable\_deletion\_protection** - (Optional) If true, deletion of the load balancer will be disabled via the AWS API. This will prevent Terraform from deleting the load balancer. Defaults to **false**.
- **enable\_cross\_zone\_load\_balancing** - (Optional) If true, cross-zone load balancing of the load balancer will be enabled. This is a **network**

load balancer feature. Defaults to **false**.

- **enable\_http2** - (Optional) Indicates whether HTTP/2 is enabled in application load balancers. Defaults to **true**.
- **ip\_address\_type** - (Optional) The type of IP addresses used by the subnets for your load balancer. The possible values are **ipv4** and **dualstack**.
- **tags** - (Optional) A mapping of tags to assign to the resource.

**NOTE::** Please note that internal LBs can only use **ipv4** as the **ip\_address\_type**. You can only change to **dualstack** **ip\_address\_type** if the selected subnets are IPv6 enabled.

Access Logs (**access\_logs**) support the following:

- **bucket** - (Required) The S3 bucket name to store the logs in.
- **prefix** - (Optional) The S3 bucket prefix. Logs are stored in the root if not configured.
- **enabled** - (Optional) Boolean to enable / disable **access\_logs**. Defaults to **false**, even when **bucket** is specified.

Subnet Mapping (**subnet\_mapping**) blocks support the following:

- **subnet\_id** - (Required) The id of the subnet of which to attach to the load balancer. You can specify only one subnet per Availability Zone.
- **allocation\_id** - (Optional) The allocation ID of the Elastic IP address.

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- **id** - The ARN of the load balancer (matches **arn**).
- **arn** - The ARN of the load balancer (matches **id**).
- **arn\_suffix** - The ARN suffix for use with CloudWatch Metrics.
- **dns\_name** - The DNS name of the load balancer.
- **canonical\_hosted\_zone\_id** - The canonical hosted zone ID of the load balancer.
- **zone\_id** - The canonical hosted zone ID of the load balancer (to be used in a Route 53 Alias record).

## » Timeouts

**aws\_lb** provides the following Timeouts configuration options:

- **create** - (Default 10 minutes) Used for Creating LB
- **update** - (Default 10 minutes) Used for LB modifications
- **delete** - (Default 10 minutes) Used for destroying LB

## » Import

LBs can be imported using their ARN, e.g.

```
$ terraform import aws_lb.bar arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer:bar
```

## » aws\_lb\_listener

Provides a Load Balancer Listener resource.

**Note:** `aws_alb_listener` is known as `aws_lb_listener`. The functionality is identical.

## » Example Usage

```
# Create a new load balancer
resource "aws_lb" "front_end" {
  # ...
}

resource "aws_lb_target_group" "front_end" {
  # ...
}

resource "aws_lb_listener" "front_end" {
  load_balancer_arn = "${aws_lb.front_end.arn}"
  port              = "443"
  protocol          = "HTTPS"
  ssl_policy        = "ELBSecurityPolicy-2015-05"
  certificate_arn    = "arn:aws:iam::187416307283:server-certificate/test_cert_rab3wuqwgja25o"

  default_action {
    target_group_arn = "${aws_lb_target_group.front_end.arn}"
    type             = "forward"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `load_balancer_arn` - (Required, Forces New Resource) The ARN of the load balancer.
- `port` - (Required) The port on which the load balancer is listening.



- **protocol** - (Optional) The protocol for connections from clients to the load balancer. Valid values are TCP, HTTP and HTTPS. Defaults to HTTP.
- **ssl\_policy** - (Optional) The name of the SSL Policy for the listener. Required if **protocol** is HTTPS.
- **certificate\_arn** - (Optional) The ARN of the default SSL server certificate. Exactly one certificate is required if the protocol is HTTPS. For adding additional SSL certificates, see the **aws\_lb\_listener\_certificate** resource.
- **default\_action** - (Required) An Action block. Action blocks are documented below.

Action Blocks (for **default\_action**) support the following:

- **target\_group\_arn** - (Required) The ARN of the Target Group to which to route traffic.
- **type** - (Required) The type of routing action. The only valid value is **forward**.

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- **id** - The ARN of the listener (matches **arn**)
- **arn** - The ARN of the listener (matches **id**)

## » Import

Listeners can be imported using their ARN, e.g.

```
$ terraform import aws_lb_listener.front_end arn:aws:elasticloadbalancing:us-west-2:18741630
```

## » aws\_lb\_listener\_certificate

Provides a Load Balancer Listener Certificate resource.

This resource is for additional certificates and does not replace the default certificate on the listener.

**Note:** **aws\_alb\_listener\_certificate** is known as **aws\_lb\_listener\_certificate**. The functionality is identical.

## » Example Usage

```
resource "aws_acm_certificate" "example" {
```

```

    # ...
}

resource "aws_lb" "front_end" {
    # ...
}

resource "aws_lb_listener" "front_end" {
    # ...
}

resource "aws_lb_listener_certificate" "example" {
    listener_arn    = "${aws_lb_listener.front_end.arn}"
    certificate_arn = "${aws_acm_certificate.example.arn}"
}

```

## » Argument Reference

The following arguments are supported:

- **listener\_arn** - (Required, Forces New Resource) The ARN of the listener to which to attach the certificate.
- **certificate\_arn** - (Required, Forces New Resource) The ARN of the certificate to attach to the listener.

## » `aws_lb_listener_rule`

Provides a Load Balancer Listener Rule resource.

**Note:** `aws_alb_listener_rule` is known as `aws_lb_listener_rule`. The functionality is identical.

## » Example Usage

```

# Create a new load balancer
resource "aws_lb" "front_end" {
    # ...
}

resource "aws_lb_listener" "front_end" {
    # Other parameters
}

```

```

resource "aws_lb_listener_rule" "static" {
  listener_arn = "${aws_lb_listener.front_end.arn}"
  priority     = 100

  action {
    type           = "forward"
    target_group_arn = "${aws_lb_target_group.static.arn}"
  }

  condition {
    field = "path-pattern"
    values = ["/static/*"]
  }
}

resource "aws_lb_listener_rule" "host_based_routing" {
  listener_arn = "${aws_lb_listener.front_end.arn}"
  priority     = 99

  action {
    type           = "forward"
    target_group_arn = "${aws_lb_target_group.static.arn}"
  }

  condition {
    field = "host-header"
    values = ["my-service.*.terraform.io"]
  }
}

```

## » Argument Reference

The following arguments are supported:

- **listener\_arn** - (Required, Forces New Resource) The ARN of the listener to which to attach the rule.
- **priority** - (Optional) The priority for the rule between 1 and 50000. Leaving it unset will automatically set the rule with next available priority after currently existing highest rule. A listener can't have multiple rules with the same priority.
- **action** - (Required) An Action block. Action blocks are documented below.
- **condition** - (Required) A Condition block. Condition blocks are documented below.

Action Blocks (for **action**) support the following:

- **target\_group\_arn** - (Required) The ARN of the Target Group to which to route traffic.
- **type** - (Required) The type of routing action. The only valid value is **forward**.

Condition Blocks (for **condition**) support the following:

- **field** - (Required) The name of the field. Must be one of **path-pattern** for path based routing or **host-header** for host based routing.
- **values** - (Required) The path patterns to match. A maximum of 1 can be defined.

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- **id** - The ARN of the rule (matches **arn**)
- **arn** - The ARN of the rule (matches **id**)

## » Import

Rules can be imported using their ARN, e.g.

```
$ terraform import aws_lb_listener_rule.front_end arn:aws:elasticloadbalancing:us-west-2:187
```

## » aws\_lb\_target\_group

Provides a Target Group resource for use with Load Balancer resources.

**Note:** `aws_alb_target_group` is known as `aws_lb_target_group`. The functionality is identical.

## » Example Usage

```
resource "aws_lb_target_group" "test" {
  name      = "tf-example-lb-tg"
  port      = 80
  protocol  = "HTTP"
  vpc_id    = "${aws_vpc.main.id}"
}

resource "aws_vpc" "main" {
  cidr_block = "10.0.0.0/16"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional, Forces new resource) The name of the target group. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**. Cannot be longer than 6 characters.
- **port** - (Required) The port on which targets receive traffic, unless overridden when registering a specific target.
- **protocol** - (Required) The protocol to use for routing traffic to the targets.
- **vpc\_id** - (Required) The identifier of the VPC in which to create the target group.
- **deregistration\_delay** - (Optional) The amount time for Elastic Load Balancing to wait before changing the state of a deregistering target from draining to unused. The range is 0-3600 seconds. The default value is 300 seconds.
- **stickiness** - (Optional) A Stickiness block. Stickiness blocks are documented below. **stickiness** is only valid if used with Load Balancers of type **Application**
- **health\_check** - (Optional) A Health Check block. Health Check blocks are documented below.
- **target\_type** - (Optional) The type of target that you must specify when registering targets with this target group. The possible values are **instance** (targets are specified by instance ID) or **ip** (targets are specified by IP address). The default is **instance**. Note that you can't specify targets for a target group using both instance IDs and IP addresses. If the target type is **ip**, specify IP addresses from the subnets of the virtual private cloud (VPC) for the target group, the RFC 1918 range (10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16), and the RFC 6598 range (100.64.0.0/10). You can't specify publicly routable IP addresses.
- **tags** - (Optional) A mapping of tags to assign to the resource.

Stickiness Blocks (**stickiness**) support the following:

- **type** - (Required) The type of sticky sessions. The only current possible value is **lb\_cookie**.
- **cookie\_duration** - (Optional) The time period, in seconds, during which requests from a client should be routed to the same target. After this time period expires, the load balancer-generated cookie is considered stale. The range is 1 second to 1 week (604800 seconds). The default value is 1 day (86400 seconds).
- **enabled** - (Optional) Boolean to enable / disable **stickiness**. Default is **true**

**NOTE:** To help facilitate the authoring of modules that support target groups

of any protocol, you can define **stickiness** regardless of the protocol chosen. However, for TCP target groups, **enabled** must be **false**.

Health Check Blocks (**health\_check**):

**Note:** The Health Check parameters you can set vary by the **protocol** of the Target Group. Many parameters cannot be set to custom values for **network** load balancers at this time. See [http://docs.aws.amazon.com/elasticloadbalancing/latest/APIReference/API\\_CreateTargetGroup.html](http://docs.aws.amazon.com/elasticloadbalancing/latest/APIReference/API_CreateTargetGroup.html) for a complete reference.

- **interval** - (Optional) The approximate amount of time, in seconds, between health checks of an individual target. Minimum value 5 seconds, Maximum value 300 seconds. Default 30 seconds.
- **path** - (Optional) The destination for the health check request. Default **/**.
- **port** - (Optional) The port to use to connect with the target. Valid values are either ports 1-65536, or **traffic-port**. Defaults to **traffic-port**.
- **protocol** - (Optional) The protocol to use to connect with the target. Defaults to HTTP.
- **timeout** - (Optional) The amount of time, in seconds, during which no response means a failed health check. For Application Load Balancers, the range is 2 to 60 seconds and the default is 5 seconds. For Network Load Balancers, you cannot set a custom value, and the default is 10 seconds for TCP and HTTPS health checks and 6 seconds for HTTP health checks.
- **healthy\_threshold** - (Optional) The number of consecutive health checks successes required before considering an unhealthy target healthy. Defaults to 3.
- **unhealthy\_threshold** - (Optional) The number of consecutive health check failures required before considering the target unhealthy. For Network Load Balancers, this value must be the same as the **healthy\_threshold**. Defaults to 3.
- **matcher** (Optional, only supported on Application Load Balancers): The HTTP codes to use when checking for a successful response from a target. You can specify multiple values (for example, "200,202") or a range of values (for example, "200-299").

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- **id** - The ARN of the Target Group (matches **arn**)
- **arn** - The ARN of the Target Group (matches **id**)
- **arn\_suffix** - The ARN suffix for use with CloudWatch Metrics.
- **name** - The name of the Target Group

## » Import

Target Groups can be imported using their ARN, e.g.

```
$ terraform import aws_lb_target_group.app_front_end arn:aws:elasticloadbalancing:us-west-2:
```

## » aws\_lb\_target\_group\_attachment

Provides the ability to register instances and containers with a LB target group

**Note:** `aws_alb_target_group_attachment` is known as `aws_lb_target_group_attachment`. The functionality is identical.

## » Example Usage

```
resource "aws_lb_target_group_attachment" "test" {
  target_group_arn = "${aws_lb_target_group.test.arn}"
  target_id       = "${aws_instance.test.id}"
  port            = 80
}

resource "aws_lb_target_group" "test" {
  // Other arguments
}

resource "aws_instance" "test" {
  // Other arguments
}
```

## » Argument Reference

The following arguments are supported:

- **target\_group\_arn** - (Required) The ARN of the target group with which to register targets
- **target\_id** (Required) The ID of the target. This is the Instance ID for an instance, or the container ID for an ECS container. If the target type is ip, specify an IP address.
- **port** - (Optional) The port on which targets receive traffic.
- **availability\_zone** - (Optional) The Availability Zone where the IP address of the target is to be registered.

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- `id` - A unique identifier for the attachment

## » Import

Target Group Attachments cannot be imported.

## » `aws_ami`

The AMI resource allows the creation and management of a completely-custom *Amazon Machine Image* (AMI).

If you just want to duplicate an existing AMI, possibly copying it to another region, it's better to use `aws_ami_copy` instead.

If you just want to share an existing AMI with another AWS account, it's better to use `aws_ami_launch_permission` instead.

## » Example Usage

```
# Create an AMI that will start a machine whose root device is backed by
# an EBS volume populated from a snapshot. It is assumed that such a snapshot
# already exists with the id "snap-xxxxxxx".
resource "aws_ami" "example" {
  name = "terraform-example"
  virtualization_type = "hvm"
  root_device_name = "/dev/xvda"

  ebs_block_device {
    device_name = "/dev/xvda"
    snapshot_id = "snap-xxxxxxx"
    volume_size = 8
  }
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) A region-unique name for the AMI.



- **description** - (Optional) A longer, human-readable description for the AMI.
- **root\_device\_name** - (Optional) The name of the root device (for example, `/dev/sda1`, or `/dev/xvda`).
- **virtualization\_type** - (Optional) Keyword to choose what virtualization mode created instances will use. Can be either "paravirtual" (the default) or "hvm". The choice of virtualization type changes the set of further arguments that are required, as described below.
- **architecture** - (Optional) Machine architecture for created instances. Defaults to "x86\_64".
- **ebs\_block\_device** - (Optional) Nested block describing an EBS block device that should be attached to created instances. The structure of this block is described below.
- **ephemeral\_block\_device** - (Optional) Nested block describing an ephemeral block device that should be attached to created instances. The structure of this block is described below.

When **virtualization\_type** is "paravirtual" the following additional arguments apply:

- **image\_location** - (Required) Path to an S3 object containing an image manifest, e.g. created by the `ec2-upload-bundle` command in the EC2 command line tools.
- **kernel\_id** - (Required) The id of the kernel image (AKI) that will be used as the paravirtual kernel in created instances.
- **ramdisk\_id** - (Optional) The id of an initrd image (ARI) that will be used when booting the created instances.

When **virtualization\_type** is "hvm" the following additional arguments apply:

- **sriov\_net\_support** - (Optional) When set to "simple" (the default), enables enhanced networking for created instances. No other value is supported at this time.

Nested **ebs\_block\_device** blocks have the following structure:

- **device\_name** - (Required) The path at which the device is exposed to created instances.
- **delete\_on\_termination** - (Optional) Boolean controlling whether the EBS volumes created to support each created instance will be deleted once that instance is terminated.
- **encrypted** - (Optional) Boolean controlling whether the created EBS volumes will be encrypted. Can't be used with **snapshot\_id**.
- **iops** - (Required only when **volume\_type** is "io1") Number of I/O operations per second the created volumes will support.
- **snapshot\_id** - (Optional) The id of an EBS snapshot that will be used to initialize the created EBS volumes. If set, the **volume\_size** attribute must be at least as large as the referenced snapshot.

- **volume\_size** - (Required unless **snapshot\_id** is set) The size of created volumes in GiB. If **snapshot\_id** is set and **volume\_size** is omitted then the volume will have the same size as the selected snapshot.
- **volume\_type** - (Optional) The type of EBS volume to create. Can be one of "standard" (the default), "io1" or "gp2".
- **kms\_key\_id** - (Optional) The full ARN of the AWS Key Management Service (AWS KMS) CMK to use when encrypting the snapshots of an image during a copy operation. This parameter is only required if you want to use a non-default CMK; if this parameter is not specified, the default CMK for EBS is used

**Note:** You can specify **encrypted** or **snapshot\_id** but not both.

Nested **ephemeral\_block\_device** blocks have the following structure:

- **device\_name** - (Required) The path at which the device is exposed to created instances.
- **virtual\_name** - (Required) A name for the ephemeral device, of the form "ephemeralN" where *N* is a volume number starting from zero.

## » Timeouts

The **timeouts** block allows you to specify timeouts for certain actions:

- **create** - (Defaults to 40 mins) Used when creating the AMI
- **update** - (Defaults to 40 mins) Used when updating the AMI
- **delete** - (Defaults to 90 mins) Used when deregistering the AMI

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the created AMI.
- **root\_snapshot\_id** - The Snapshot ID for the root volume (for EBS-backed AMIs)

## » aws\_\_ami\_\_copy

The "AMI copy" resource allows duplication of an Amazon Machine Image (AMI), including cross-region copies.

If the source AMI has associated EBS snapshots, those will also be duplicated along with the AMI.

This is useful for taking a single AMI provisioned in one region and making it available in another for a multi-region deployment.

Copying an AMI can take several minutes. The creation of this resource will block until the new AMI is available for use on new instances.

## » Example Usage

```
resource "aws_ami_copy" "example" {
  name           = "terraform-example"
  description    = "A copy of ami-xxxxxxx"
  source_ami_id  = "ami-xxxxxxx"
  source_ami_region = "us-west-1"

  tags {
    Name = "HelloWorld"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A region-unique name for the AMI.
- **source\_ami\_id** - (Required) The id of the AMI to copy. This id must be valid in the region given by **source\_ami\_region**.
- **source\_ami\_region** - (Required) The region from which the AMI will be copied. This may be the same as the AWS provider region in order to create a copy within the same region.
- **encrypted** - (Optional) Specifies whether the destination snapshots of the copied image should be encrypted. Defaults to **false**
- **kms\_key\_id** - (Optional) The full ARN of the KMS Key to use when encrypting the snapshots of an image during a copy operation. If not specified, then the default AWS KMS Key will be used

This resource also exposes the full set of arguments from the **aws\_ami** resource.

## » Timeouts

The **timeouts** block allows you to specify timeouts for certain actions:

- **create** - (Defaults to 40 mins) Used when creating the AMI
- **update** - (Defaults to 40 mins) Used when updating the AMI
- **delete** - (Defaults to 90 mins) Used when deregistering the AMI

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the created AMI.

This resource also exports a full set of attributes corresponding to the arguments of the `aws_ami` resource, allowing the properties of the created AMI to be used elsewhere in the configuration.

## » `aws_ami_from_instance`

The "AMI from instance" resource allows the creation of an Amazon Machine Image (AMI) modelled after an existing EBS-backed EC2 instance.

The created AMI will refer to implicitly-created snapshots of the instance's EBS volumes and mimic its assigned block device configuration at the time the resource is created.

This resource is best applied to an instance that is stopped when this instance is created, so that the contents of the created image are predictable. When applied to an instance that is running, *the instance will be stopped before taking the snapshots and then started back up again*, resulting in a period of downtime.

Note that the source instance is inspected only at the initial creation of this resource. Ongoing updates to the referenced instance will not be propagated into the generated AMI. Users may taint or otherwise recreate the resource in order to produce a fresh snapshot.

## » Example Usage

```
resource "aws_ami_from_instance" "example" {
  name           = "terraform-example"
  source_instance_id = "i-xxxxxxx"
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) A region-unique name for the AMI.
- `source_instance_id` - (Required) The id of the instance to use as the basis of the AMI.

- `snapshot_without_reboot` - (Optional) Boolean that overrides the behavior of stopping the instance before snapshotting. This is risky since it may cause a snapshot of an inconsistent filesystem state, but can be used to avoid downtime if the user otherwise guarantees that no filesystem writes will be underway at the time of snapshot.

## » Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the created AMI.

This resource also exports a full set of attributes corresponding to the arguments of the `aws_ami` resource, allowing the properties of the created AMI to be used elsewhere in the configuration.

## » `aws_ami_launch_permission`

Adds launch permission to Amazon Machine Image (AMI) from another AWS account.

## » Example Usage

```
resource "aws_ami_launch_permission" "example" {
  image_id   = "ami-12345678"
  account_id = "123456789012"
}
```

## » Argument Reference

The following arguments are supported:

- `image_id` - (required) A region-unique name for the AMI.
- `account_id` - (required) An AWS Account ID to add launch permissions.

## » Attributes Reference

The following attributes are exported:

- `id` - A combination of "image\_id-account\_id".

## » `aws_app_cookie_stickiness_policy`

Provides an application cookie stickiness policy, which allows an ELB to wed its sticky cookie's expiration to a cookie generated by your application.

## » Example Usage

```
resource "aws_elb" "lb" {
  name           = "test-lb"
  availability_zones = ["us-east-1a"]

  listener {
    instance_port      = 8000
    instance_protocol = "http"
    lb_port            = 80
    lb_protocol        = "http"
  }
}

resource "aws_app_cookie_stickiness_policy" "foo" {
  name           = "foo_policy"
  load_balancer = "${aws_elb.lb.name}"
  lb_port       = 80
  cookie_name   = "MyAppCookie"
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the stickiness policy.
- `load_balancer` - (Required) The name of load balancer to which the policy should be attached.
- `lb_port` - (Required) The load balancer port to which the policy should be applied. This must be an active listener on the load balancer.
- `cookie_name` - (Required) The application cookie whose lifetime the ELB's cookie should follow.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the policy.
- `name` - The name of the stickiness policy.
- `load_balancer` - The name of load balancer to which the policy is attached.
- `lb_port` - The load balancer port to which the policy is applied.
- `cookie_name` - The application cookie whose lifetime the ELB's cookie should follow.

## » `aws_autoscaling_attachment`

Provides an AutoScaling Attachment resource.

**NOTE on AutoScaling Groups and ASG Attachments:** Terraform currently provides both a standalone ASG Attachment resource (describing an ASG attached to an ELB), and an AutoScaling Group resource with `load_balancers` defined in-line. At this time you cannot use an ASG with in-line load balancers in conjunction with an ASG Attachment resource. Doing so will cause a conflict and will overwrite attachments.

## » Example Usage

```
# Create a new load balancer attachment
resource "aws_autoscaling_attachment" "asg_attachment_bar" {
  autoscaling_group_name = "${aws_autoscaling_group.asg.id}"
  elb                    = "${aws_elb.bar.id}"
}

# Create a new ALB Target Group attachment
resource "aws_autoscaling_attachment" "asg_attachment_bar" {
  autoscaling_group_name = "${aws_autoscaling_group.asg.id}"
  alb_target_group_arn   = "${aws_alb_target_group.test.arn}"
}
```

## » Argument Reference

The following arguments are supported:

- `autoscaling_group_name` - (Required) Name of ASG to associate with the ELB.
- `elb` - (Optional) The name of the ELB.

- `alb_target_group_arn` - (Optional) The ARN of an ALB Target Group.

## » `aws__autoscaling_group`

Provides an AutoScaling Group resource.

**Note:** You must specify either `launch_configuration` or `launch_template`.

### » Example Usage

```
resource "aws_placement_group" "test" {
  name      = "test"
  strategy  = "cluster"
}

resource "aws_autoscaling_group" "bar" {
  name                  = "foobar3-terraform-test"
  max_size              = 5
  min_size              = 2
  health_check_grace_period = 300
  health_check_type     = "ELB"
  desired_capacity       = 4
  force_delete           = true
  placement_group        = "${aws_placement_group.test.id}"
  launch_configuration    = "${aws_launch_configuration.foobar.name}"
  vpc_zone_identifier     = ["${aws_subnet.example1.id}", "${aws_subnet.example2.id}"]

  initial_lifecycle_hook {
    name              = "foobar"
    default_result    = "CONTINUE"
    heartbeat_timeout = 2000
    lifecycle_transition = "autoscaling:EC2_INSTANCE_LAUNCHING"

    notification_metadata = <<EOF
{
  "foo": "bar"
}
EOF

    notification_target_arn = "arn:aws:sqs:us-east-1:444455556666:queue1*"
    role_arn                 = "arn:aws:iam::123456789012:role/S3Access"
  }

  tag {
```



```

    key          = "foo"
    value        = "bar"
    propagate_at_launch = true
  }

  timeouts {
    delete = "15m"
  }

  tag {
    key          = "lorem"
    value        = "ipsum"
    propagate_at_launch = false
  }
}

```

#### » With Latest Version Of Launch Template

```

resource "aws_launch_template" "foobar" {
  name_prefix = "foobar"
  image_id    = "ami-1a2b3c"
  instance_type = "t2.micro"
}

resource "aws_autoscaling_group" "bar" {
  availability_zones = ["us-east-1a"]
  desired_capacity   = 1
  max_size           = 1
  min_size           = 1
  launch_template = {
    id = "${aws_launch_template.foobar.id}"
    version = "$$Latest"
  }
}

```

#### » Interpolated tags

```

variable extra_tags {
  default = [
    {
      key = "Foo"
      value = "Bar"
      propagate_at_launch = true
    },
  ]
}

```

```

    {
      key = "Baz"
      value = "Bam"
      propagate_at_launch = true
    },
  ]
}

resource "aws_autoscaling_group" "bar" {
  name           = "foobar3-terraform-test"
  max_size       = 5
  min_size       = 2
  launch_configuration = "${aws_launch_configuration.foobar.name}"
  vpc_zone_identifier = ["${aws_subnet.example1.id}", "${aws_subnet.example2.id}"]

  tags = [
    {
      key           = "explicit1"
      value         = "value1"
      propagate_at_launch = true
    },
    {
      key           = "explicit2"
      value         = "value2"
      propagate_at_launch = true
    },
  ]

  tags = ["${concat(
    list(
      map("key", "interpolation1", "value", "value3", "propagate_at_launch", true),
      map("key", "interpolation2", "value", "value4", "propagate_at_launch", true)
    ),
    var.extra_tags
  )}"]
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The name of the auto scaling group. By default generated by Terraform.
- **name\_prefix** - (Optional) Creates a unique name beginning with the specified prefix. Conflicts with **name**.

- **max\_size** - (Required) The maximum size of the auto scale group.
- **min\_size** - (Required) The minimum size of the auto scale group. (See also Waiting for Capacity below.)
- **availability\_zones** - (Required only for EC2-Classic) A list of one or more availability zones for the group. This parameter should not be specified when using **vpc\_zone\_identifier**.
- **default\_cooldown** - (Optional) The amount of time, in seconds, after a scaling activity completes before another scaling activity can start.
- **launch\_configuration** - (Optional) The name of the launch configuration to use.
- **launch\_template** - (Optional) Launch template specification to use to launch instances. See Launch Template Specification below for more details.
- **initial\_lifecycle\_hook** - (Optional) One or more Lifecycle Hooks to attach to the autoscaling group **before** instances are launched. The syntax is exactly the same as the separate **aws\_autoscaling\_lifecycle\_hook** resource, without the **autoscaling\_group\_name** attribute. Please note that this will only work when creating a new autoscaling group. For all other use-cases, please use **aws\_autoscaling\_lifecycle\_hook** resource.
- **health\_check\_grace\_period** - (Optional, Default: 300) Time (in seconds) after instance comes into service before checking health.
- **health\_check\_type** - (Optional) "EC2" or "ELB". Controls how health checking is done.
- **desired\_capacity** - (Optional) The number of Amazon EC2 instances that should be running in the group. (See also Waiting for Capacity below.)
- **force\_delete** - (Optional) Allows deleting the autoscaling group without waiting for all instances in the pool to terminate. You can force an autoscaling group to delete even if it's in the process of scaling a resource. Normally, Terraform drains all the instances before deleting the group. This bypasses that behavior and potentially leaves resources dangling.
- **load\_balancers** (Optional) A list of elastic load balancer names to add to the autoscaling group names. Only valid for classic load balancers. For ALBs, use **target\_group\_arns** instead.
- **vpc\_zone\_identifier** (Optional) A list of subnet IDs to launch resources in.
- **target\_group\_arns** (Optional) A list of **aws\_alb\_target\_group** ARNs, for use with Application Load Balancing.
- **termination\_policies** (Optional) A list of policies to decide how the instances in the auto scale group should be terminated. The allowed values are **OldestInstance**, **NewestInstance**, **OldestLaunchConfiguration**, **ClosestToNextInstanceHour**, **Default**.
- **suspended\_processes** - (Optional) A list of processes to suspend for the AutoScaling Group. The allowed values are **Launch**, **Terminate**, **HealthCheck**, **ReplaceUnhealthy**, **AZRebalance**, **AlarmNotification**, **ScheduledActions**, **AddToLoadBalancer**. Note that if you suspend either

the `Launch` or `Terminate` process types, it can prevent your autoscaling group from functioning properly.

- **tag** (Optional) A list of tag blocks. Tags documented below.
- **tags** (Optional) A list of tag blocks (maps). Tags documented below.
- **placement\_group** (Optional) The name of the placement group into which you'll launch your instances, if any.
- **metrics\_granularity** - (Optional) The granularity to associate with the metrics to collect. The only valid value is `1Minute`. Default is `1Minute`.
- **enabled\_metrics** - (Optional) A list of metrics to collect. The allowed values are `GroupMinSize`, `GroupMaxSize`, `GroupDesiredCapacity`, `GroupInServiceInstances`, `GroupPendingInstances`, `GroupStandbyInstances`, `GroupTerminatingInstances`, `GroupTotalInstances`.
- **wait\_for\_capacity\_timeout** (Default: "10m") A maximum duration that Terraform should wait for ASG instances to be healthy before timing out. (See also `Waiting for Capacity` below.) Setting this to "0" causes Terraform to skip all Capacity Waiting behavior.
- **min\_elb\_capacity** - (Optional) Setting this causes Terraform to wait for this number of instances to show up healthy in the ELB only on creation. Updates will not wait on ELB instance number changes. (See also `Waiting for Capacity` below.)
- **wait\_for\_elb\_capacity** - (Optional) Setting this will cause Terraform to wait for exactly this number of healthy instances in all attached load balancers on both create and update operations. (Takes precedence over `min_elb_capacity` behavior.) (See also `Waiting for Capacity` below.)
- **protect\_from\_scale\_in** (Optional) Allows setting instance protection. The autoscaling group will not select instances with this setting for termination during scale in events.
- **service\_linked\_role\_arn** (Optional) The ARN of the service-linked role that the ASG will use to call other AWS services

Tags support the following:

The **tag** attribute accepts exactly one tag declaration with the following fields:

- **key** - (Required) Key
- **value** - (Required) Value
- **propagate\_at\_launch** - (Required) Enables propagation of the tag to Amazon EC2 instances launched via this ASG

To declare multiple tags additional **tag** blocks can be specified. Alternatively the **tags** attributes can be used, which accepts a list of maps containing the above field names as keys and their respective values. This allows the construction of dynamic lists of tags which is not possible using the single **tag** attribute. **tag** and **tags** are mutually exclusive, only one of them can be specified.

## » Launch Template Specification

The `launch_template` block supports the following:

- `id` - The ID of the launch template. Conflicts with `name`.
- `name` - The name of the launch template. Conflicts with `id`.
- `version` - Template version. Can be version number, `$Latest` or `$Default`. (Default: `$Default`).

## » Attributes Reference

The following attributes are exported:

- `id` - The autoscaling group id.
- `arn` - The ARN for this AutoScaling Group
- `availability_zones` - The availability zones of the autoscale group.
- `min_size` - The minimum size of the autoscale group
- `max_size` - The maximum size of the autoscale group
- `default_cooldown` - Time between a scaling activity and the succeeding scaling activity.
- `name` - The name of the autoscale group
- `health_check_grace_period` - Time after instance comes into service before checking health.
- `health_check_type` - "EC2" or "ELB". Controls how health checking is done.
- `desired_capacity` - The number of Amazon EC2 instances that should be running in the group.
- `launch_configuration` - The launch configuration of the autoscale group
- `vpc_zone_identifier` (Optional) - The VPC zone identifier
- `load_balancers` (Optional) The load balancer names associated with the autoscaling group.
- `target_group_arns` (Optional) list of Target Group ARNs that apply to this AutoScaling Group

**NOTE:** When using ELB as the `health_check_type`, `health_check_grace_period` is required.

**NOTE:** Terraform has two types of ways you can add lifecycle hooks - via the `initial_lifecycle_hook` attribute from this resource, or via the separate `aws_autoscaling_lifecycle_hook` resource. `initial_lifecycle_hook` exists here because any lifecycle hooks added with `aws_autoscaling_lifecycle_hook` will not be added until the autoscaling group has been created, and depending on your capacity settings, after the initial instances have been launched, creating unintended behavior. If you need hooks to run on all instances, add them with `initial_lifecycle_hook` here, but take care to not duplicate these hooks in `aws_autoscaling_lifecycle_hook`.

## » Timeouts

`autoscaling_group` provides the following Timeouts configuration options:

- `delete` - (Default 10 minutes) Used for destroying ASG.

## » Waiting for Capacity

A newly-created ASG is initially empty and begins to scale to `min_size` (or `desired_capacity`, if specified) by launching instances using the provided Launch Configuration. These instances take time to launch and boot.

On ASG Update, changes to these values also take time to result in the target number of instances providing service.

Terraform provides two mechanisms to help consistently manage ASG scale up time across dependent resources.

### » Waiting for ASG Capacity

The first is default behavior. Terraform waits after ASG creation for `min_size` (or `desired_capacity`, if specified) healthy instances to show up in the ASG before continuing.

If `min_size` or `desired_capacity` are changed in a subsequent update, Terraform will also wait for the correct number of healthy instances before continuing.

Terraform considers an instance "healthy" when the ASG reports `HealthStatus: "Healthy"` and `LifecycleState: "InService"`. See the AWS AutoScaling Docs for more information on an ASG's lifecycle.

Terraform will wait for healthy instances for up to `wait_for_capacity_timeout`. If ASG creation is taking more than a few minutes, it's worth investigating for scaling activity errors, which can be caused by problems with the selected Launch Configuration.

Setting `wait_for_capacity_timeout` to "0" disables ASG Capacity waiting.

### » Waiting for ELB Capacity

The second mechanism is optional, and affects ASGs with attached ELBs specified via the `load_balancers` attribute.

The `min_elb_capacity` parameter causes Terraform to wait for at least the requested number of instances to show up "InService" in all attached ELBs during ASG creation. It has no effect on ASG updates.

If `wait_for_elb_capacity` is set, Terraform will wait for exactly that number of Instances to be "InService" in all attached ELBs on both creation and updates.

These parameters can be used to ensure that service is being provided before Terraform moves on. If new instances don't pass the ELB's health checks for any reason, the Terraform apply will time out, and the ASG will be marked as tainted (i.e. marked to be destroyed in a follow up run).

As with ASG Capacity, Terraform will wait for up to `wait_for_capacity_timeout` for the proper number of instances to be healthy.

### » Troubleshooting Capacity Waiting Timeouts

If ASG creation takes more than a few minutes, this could indicate one of a number of configuration problems. See the AWS Docs on Load Balancer Troubleshooting for more information.

### » Import

AutoScaling Groups can be imported using the `name`, e.g.

```
$ terraform import aws_autoscaling_group.web web-asg
```

### » `aws__autoscaling__lifecycle__hook`

Provides an AutoScaling Lifecycle Hook resource.

**NOTE:** Terraform has two types of ways you can add lifecycle hooks - via the `initial_lifecycle_hook` attribute from the `aws_autoscaling_group` resource, or via this one. Hooks added via this resource will not be added until the autoscaling group has been created, and depending on your capacity settings, after the initial instances have been launched, creating unintended behavior. If you need hooks to run on all instances, add them with `initial_lifecycle_hook` in `aws_autoscaling_group`, but take care to not duplicate those hooks with this resource.

### » Example Usage

```
resource "aws_autoscaling_group" "foobar" {
  availability_zones = ["us-west-2a"]
  name               = "terraform-test-foobar5"
  health_check_type  = "EC2"
  termination_policies = ["OldestInstance"]
}
```

```

tag {
  key          = "Foo"
  value        = "foo-bar"
  propagate_at_launch = true
}

resource "aws_autoscaling_lifecycle_hook" "foobar" {
  name                  = "foobar"
  autoscaling_group_name = "${aws_autoscaling_group.foobar.name}"
  default_result        = "CONTINUE"
  heartbeat_timeout     = 2000
  lifecycle_transition  = "autoscaling:EC2_INSTANCE_LAUNCHING"

  notification_metadata = <<EOF
{
  "foo": "bar"
}
EOF

  notification_target_arn = "arn:aws:sqs:us-east-1:444455556666:queue1*"
  role_arn                = "arn:aws:iam::123456789012:role/S3Access"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the lifecycle hook.
- **autoscaling\_group\_name** - (Required) The name of the Auto Scaling group to which you want to assign the lifecycle hook
- **default\_result** - (Optional) Defines the action the Auto Scaling group should take when the lifecycle hook timeout elapses or if an unexpected failure occurs. The value for this parameter can be either CONTINUE or ABANDON. The default value for this parameter is ABANDON.
- **heartbeat\_timeout** - (Optional) Defines the amount of time, in seconds, that can elapse before the lifecycle hook times out. When the lifecycle hook times out, Auto Scaling performs the action defined in the DefaultResult parameter
- **lifecycle\_transition** - (Required) The instance state to which you want to attach the lifecycle hook. For a list of lifecycle hook types, see [describe-lifecycle-hook-types](#)
- **notification\_metadata** - (Optional) Contains additional information that you want to include any time Auto Scaling sends a message to the



notification target.

- **notification\_target\_arn** - (Optional) The ARN of the notification target that Auto Scaling will use to notify you when an instance is in the transition state for the lifecycle hook. This ARN target can be either an SQS queue or an SNS topic.
- **role\_arn** - (Optional) The ARN of the IAM role that allows the Auto Scaling group to publish to the specified notification target.

## » **aws\_\_autoscaling\_\_notification**

Provides an AutoScaling Group with Notification support, via SNS Topics. Each of the **notifications** map to a Notification Configuration inside Amazon Web Services, and are applied to each AutoScaling Group you supply.

### » **Example Usage**

Basic usage:

```
resource "aws_autoscaling_notification" "example_notifications" {
  group_names = [
    "${aws_autoscaling_group.bar.name}",
    "${aws_autoscaling_group.foo.name}",
  ]

  notifications = [
    "autoscaling:EC2_INSTANCE_LAUNCH",
    "autoscaling:EC2_INSTANCE_TERMINATE",
    "autoscaling:EC2_INSTANCE_LAUNCH_ERROR",
  ]

  topic_arn = "${aws_sns_topic.example.arn}"
}

resource "aws_sns_topic" "example" {
  name = "example-topic"

  # arn is an exported attribute
}

resource "aws_autoscaling_group" "bar" {
  name = "foobar1-terraform-test"

  # ...
}
```

```
resource "aws_autoscaling_group" "foo" {
  name = "barfoo-terraform-test"

  # ...
}
```

## » Argument Reference

The following arguments are supported:

- **group\_names** - (Required) A list of AutoScaling Group Names
- **notifications** - (Required) A list of Notification Types that trigger notifications. Acceptable values are documented in the AWS documentation [here](#)
- **topic\_arn** - (Required) The Topic ARN for notifications to be sent through

## » Attributes Reference

The following attributes are exported:

- **group\_names**
- **notifications**
- **topic\_arn**

## » `aws__autoscaling__policy`

Provides an AutoScaling Scaling Policy resource.

**NOTE:** You may want to omit `desired_capacity` attribute from attached `aws_autoscaling_group` when using autoscaling policies. It's good practice to pick either manual or dynamic (policy-based) scaling.

## » Example Usage

```
resource "aws_autoscaling_policy" "bat" {
  name                = "foobar3-terraform-test"
  scaling_adjustment  = 4
  adjustment_type     = "ChangeInCapacity"
  cooldown            = 300
  autoscaling_group_name = "${aws_autoscaling_group.bar.name}"
}
```

```

resource "aws_autoscaling_group" "bar" {
  availability_zones = ["us-east-1a"]
  name               = "foobar3-terraform-test"
  max_size           = 5
  min_size           = 2
  health_check_grace_period = 300
  health_check_type   = "ELB"
  force_delete        = true
  launch_configuration = "${aws_launch_configuration.foo.name}"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the policy.
- **autoscaling\_group\_name** - (Required) The name of the autoscaling group.
- **adjustment\_type** - (Optional) Specifies whether the adjustment is an absolute number or a percentage of the current capacity. Valid values are `ChangeInCapacity`, `ExactCapacity`, and `PercentChangeInCapacity`.
- **policy\_type** - (Optional) The policy type, either `"SimpleScaling"`, `"StepScaling"` or `"TargetTrackingScaling"`. If this value isn't provided, AWS will default to `"SimpleScaling"`.

The following arguments are only available to `"SimpleScaling"` type policies:

- **cooldown** - (Optional) The amount of time, in seconds, after a scaling activity completes and before the next scaling activity can start.
- **scaling\_adjustment** - (Optional) The number of instances by which to scale. **adjustment\_type** determines the interpretation of this number (e.g., as an absolute number or as a percentage of the existing Auto Scaling group size). A positive increment adds to the current capacity and a negative value removes from the current capacity.

The following arguments are only available to `"StepScaling"` type policies:

- **metric\_aggregation\_type** - (Optional) The aggregation type for the policy's metrics. Valid values are `"Minimum"`, `"Maximum"`, and `"Average"`. Without a value, AWS will treat the aggregation type as `"Average"`.
- **estimated\_instance\_warmup** - (Optional) The estimated time, in seconds, until a newly launched instance will contribute CloudWatch metrics. Without a value, AWS will default to the group's specified cooldown period.
- **step\_adjustments** - (Optional) A set of adjustments that manage group scaling. These have the following structure:

```

step_adjustment {
  scaling_adjustment = -1
  metric_interval_lower_bound = 1.0
  metric_interval_upper_bound = 2.0
}
step_adjustment {
  scaling_adjustment = 1
  metric_interval_lower_bound = 2.0
  metric_interval_upper_bound = 3.0
}

```

The following fields are available in step adjustments:

- **scaling\_adjustment** - (Required) The number of members by which to scale, when the adjustment bounds are breached. A positive value scales up. A negative value scales down.
- **metric\_interval\_lower\_bound** - (Optional) The lower bound for the difference between the alarm threshold and the CloudWatch metric. Without a value, AWS will treat this bound as infinity.
- **metric\_interval\_upper\_bound** - (Optional) The upper bound for the difference between the alarm threshold and the CloudWatch metric. Without a value, AWS will treat this bound as infinity. The upper bound must be greater than the lower bound.

The following arguments are only available to "TargetTrackingScaling" type policies:

- **target\_tracking\_configuration** - (Optional) A target tracking policy. These have the following structure:

```

target_tracking_configuration {
  predefined_metric_specification {
    predefined_metric_type = "ASGAverageCPUUtilization"
  }
  target_value = 40.0
}
target_tracking_configuration {
  customized_metric_specification {
    metric_dimension {
      name = "fuga"
      value = "fuga"
    }
    metric_name = "hoge"
    namespace = "hoge"
    statistic = "Average"
  }
  target_value = 40.0
}

```

The following fields are available in target tracking configuration:

- **predefined\_metric\_specification** - (Optional) A predefined metric. Conflicts with **customized\_metric\_specification**.
- **customized\_metric\_specification** - (Optional) A customized metric. Conflicts with **predefined\_metric\_specification**.
- **target\_value** - (Required) The target value for the metric.
- **disable\_scale\_in** - (Optional, Default: false) Indicates whether scale in by the target tracking policy is disabled.

#### » **predefined\_\_metric\_\_specification**

The following arguments are supported:

- **predefined\_metric\_type** - (Required) The metric type.
- **resource\_label** - (Optional) Identifies the resource associated with the metric type.

#### » **customized\_\_metric\_\_specification**

The following arguments are supported:

- **metric\_dimension** - (Optional) The dimensions of the metric.
- **metric\_name** - (Required) The name of the metric.
- **namespace** - (Required) The namespace of the metric.
- **statistic** - (Required) The statistic of the metric.
- **unit** - (Optional) The unit of the metric.

#### » **metric\_\_dimension**

The following arguments are supported:

- **name** - (Required) The name of the dimension.
- **value** - (Required) The value of the dimension.

The following arguments are supported for backwards compatibility but should not be used:

- **min\_adjustment\_step** - (Optional) Use **min\_adjustment\_magnitude** instead.

#### » **Attribute Reference**

- **arn** - The ARN assigned by AWS to the scaling policy.
- **name** - The scaling policy's name.

- `autoscaling_group_name` - The scaling policy's assigned autoscaling group.
- `adjustment_type` - The scaling policy's adjustment type.
- `policy_type` - The scaling policy's type.

## » `aws__autoscaling__schedule`

Provides an AutoScaling Schedule resource.

### » Example Usage

```
resource "aws_autoscaling_group" "foobar" {
  availability_zones = ["us-west-2a"]
  name               = "terraform-test-foobar5"
  max_size           = 1
  min_size           = 1
  health_check_grace_period = 300
  health_check_type   = "ELB"
  force_delete        = true
  termination_policies = ["OldestInstance"]
}

resource "aws_autoscaling_schedule" "foobar" {
  scheduled_action_name = "foobar"
  min_size              = 0
  max_size              = 1
  desired_capacity      = 0
  start_time            = "2016-12-11T18:00:00Z"
  end_time              = "2016-12-12T06:00:00Z"
  autoscaling_group_name = "${aws_autoscaling_group.foobar.name}"
}
```

### » Argument Reference

The following arguments are supported:

- `autoscaling_group_name` - (Required) The name or Amazon Resource Name (ARN) of the Auto Scaling group.
- `scheduled_action_name` - (Required) The name of this scaling action.
- `start_time` - (Optional) The time for this action to start, in "YYYY-MM-DDThh:mm:ssZ" format in UTC/GMT only (for example, 2014-06-01T00:00:00Z ). If you try to schedule your action in the past, Auto Scaling returns an error message.

- **end\_time** - (Optional) The time for this action to end, in "YYYY-MM-DDThh:mm:ssZ" format in UTC/GMT only (for example, 2014-06-01T00:00:00Z ). If you try to schedule your action in the past, Auto Scaling returns an error message.
- **recurrence** - (Optional) The time when recurring future actions will start. Start time is specified by the user following the Unix cron syntax format.
- **min\_size** - (Optional) The minimum size for the Auto Scaling group. Default 0. Set to -1 if you don't want to change the minimum size at the scheduled time.
- **max\_size** - (Optional) The maximum size for the Auto Scaling group. Default 0. Set to -1 if you don't want to change the maximum size at the scheduled time.
- **desired\_capacity** - (Optional) The number of EC2 instances that should be running in the group. Default 0. Set to -1 if you don't want to change the desired capacity at the scheduled time.

**NOTE:** When **start\_time** and **end\_time** are specified with **recurrence** , they form the boundaries of when the recurring action will start and stop.

## » Attribute Reference

- **arn** - The ARN assigned by AWS to the autoscaling schedule.

## » **aws\_\_snapshot\_\_create\_\_volume\_\_permission**

Adds permission to create volumes off of a given EBS Snapshot.

## » Example Usage

```
resource "aws_snapshot_create_volume_permission" "example_perm" {
  snapshot_id = "${aws_ebs_snapshot.example_snapshot.id}"
  account_id  = "12345678"
}

resource "aws_ebs_volume" "example" {
  availability_zone = "us-west-2a"
  size              = 40
}

resource "aws_ebs_snapshot" "example_snapshot" {
  volume_id = "${aws_ebs_volume.example.id}"
}
```

## » Argument Reference

The following arguments are supported:

- **snapshot\_id** - (required) A snapshot ID
- **account\_id** - (required) An AWS Account ID to add create volume permissions

## » Attributes Reference

The following attributes are exported:

- **id** - A combination of "snapshot\_id-account\_id".

## » aws\_ebs\_snapshot

Creates a Snapshot of an EBS Volume.

## » Example Usage

```
resource "aws_ebs_volume" "example" {
  availability_zone = "us-west-2a"
  size = 40
  tags {
    Name = "HelloWorld"
  }
}

resource "aws_ebs_snapshot" "example_snapshot" {
  volume_id = "${aws_ebs_volume.example.id}"

  tags {
    Name = "HelloWorld_snap"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **volume\_id** - (Required) The Volume ID of which to make a snapshot.
- **description** - (Optional) A description of what the snapshot is.
- **tags** - (Optional) A mapping of tags to assign to the snapshot



## » Attributes Reference

The following attributes are exported:

- `id` - The snapshot ID (e.g. `snap-59fcb34e`).
- `owner_id` - The AWS account ID of the EBS snapshot owner.
- `owner_alias` - Value from an Amazon-maintained list (`amazon`, `aws-marketplace`, `microsoft`) of snapshot owners.
- `encrypted` - Whether the snapshot is encrypted.
- `volume_size` - The size of the drive in GiBs.
- `kms_key_id` - The ARN for the KMS encryption key.
- `data_encryption_key_id` - The data encryption key identifier for the snapshot.
- `tags` - A mapping of tags for the snapshot.

## » `aws_ebs_volume`

Manages a single EBS volume.

## » Example Usage

```
resource "aws_ebs_volume" "example" {
  availability_zone = "us-west-2a"
  size = 40
  tags {
    Name = "HelloWorld"
  }
}
```

**NOTE:** One of `size` or `snapshot_id` is required when specifying an EBS volume

## » Argument Reference

The following arguments are supported:

- `availability_zone` - (Required) The AZ where the EBS volume will exist.
- `encrypted` - (Optional) If true, the disk will be encrypted.
- `iops` - (Optional) The amount of IOPS to provision for the disk.
- `size` - (Optional) The size of the drive in GiBs.
- `snapshot_id` (Optional) A snapshot to base the EBS volume off of.
- `type` - (Optional) The type of EBS volume. Can be `"standard"`, `"gp2"`, `"io1"`, `"sc1"` or `"st1"` (Default: `"standard"`).

- `kms_key_id` - (Optional) The ARN for the KMS encryption key. When specifying `kms_key_id`, `encrypted` needs to be set to true.
- `tags` - (Optional) A mapping of tags to assign to the resource.

**NOTE:** When changing the `size`, `iops` or `type` of an instance, there are considerations to be aware of that Amazon have written about this.

## » Attributes Reference

The following attributes are exported:

- `id` - The volume ID (e.g. `vol-59fcb34e`).
- `arn` - The volume ARN (e.g. `arn:aws:ec2:us-east-1:0123456789012:volume/vol-59fcb34e`).

## » Import

EBS Volumes can be imported using the `id`, e.g.

```
$ terraform import aws_ebs_volume.data vol-049df61146c4d7901
```

## » aws\_\_eip

Provides an Elastic IP resource.

**Note:** EIP may require IGW to exist prior to association. Use `depends_on` to set an explicit dependency on the IGW.

## » Example Usage

Single EIP associated with an instance:

```
resource "aws_eip" "lb" {
  instance = "${aws_instance.web.id}"
  vpc      = true
}
```

Multiple EIPs associated with a single network interface:

```
resource "aws_network_interface" "multi-ip" {
  subnet_id = "${aws_subnet.main.id}"
  private_ips = ["10.0.0.10", "10.0.0.11"]
}

resource "aws_eip" "one" {
```

```

    vpc                        = true
    network_interface          = "${aws_network_interface.multi-ip.id}"
    associate_with_private_ip = "10.0.0.10"
}

```

```

resource "aws_eip" "two" {
    vpc                        = true
    network_interface          = "${aws_network_interface.multi-ip.id}"
    associate_with_private_ip = "10.0.0.11"
}

```

Attaching an EIP to an Instance with a pre-assigned private ip (VPC Only):

```

resource "aws_vpc" "default" {
    cidr_block      = "10.0.0.0/16"
    enable_dns_hostnames = true
}

```

```

resource "aws_internet_gateway" "gw" {
    vpc_id = "${aws_vpc.default.id}"
}

```

```

resource "aws_subnet" "tf_test_subnet" {
    vpc_id            = "${aws_vpc.default.id}"
    cidr_block        = "10.0.0.0/24"
    map_public_ip_on_launch = true

    depends_on = ["aws_internet_gateway.gw"]
}

```

```

resource "aws_instance" "foo" {
    # us-west-2
    ami          = "ami-5189a661"
    instance_type = "t2.micro"

    private_ip = "10.0.0.12"
    subnet_id   = "${aws_subnet.tf_test_subnet.id}"
}

```

```

resource "aws_eip" "bar" {
    vpc = true

    instance          = "${aws_instance.foo.id}"
    associate_with_private_ip = "10.0.0.12"
    depends_on        = ["aws_internet_gateway.gw"]
}

```

## » Argument Reference

The following arguments are supported:

- **vpc** - (Optional) Boolean if the EIP is in a VPC or not.
- **instance** - (Optional) EC2 instance ID.
- **network\_interface** - (Optional) Network interface ID to associate with.
- **associate\_with\_private\_ip** - (Optional) A user specified primary or secondary private IP address to associate with the Elastic IP address. If no private IP address is specified, the Elastic IP address is associated with the primary private IP address.
- **tags** - (Optional) A mapping of tags to assign to the resource.

**NOTE:** You can specify either the **instance** ID or the **network\_interface** ID, but not both. Including both will **not** return an error from the AWS API, but will have undefined behavior. See the relevant AssociateAddress API Call for more information.

## » Attributes Reference

The following additional attributes are exported:

- **id** - Contains the EIP allocation ID.
- **private\_ip** - Contains the private IP address (if in VPC).
- **associate\_with\_private\_ip** - Contains the user specified private IP address (if in VPC).
- **public\_ip** - Contains the public IP address.
- **instance** - Contains the ID of the attached instance.
- **network\_interface** - Contains the ID of the attached network interface.

## » Timeouts

**aws\_eip** provides the following Timeouts configuration options:

- **read** - (Default 15 minutes) How long to wait querying for information about EIPs.
- **update** - (Default 5 minutes) How long to wait for an EIP to be updated.
- **delete** - (Default 3 minutes) How long to wait for an EIP to be deleted.

## » Import

EIPs in a VPC can be imported using their Allocation ID, e.g.

```
$ terraform import aws_eip.bar eipalloc-00a10e96
```

EIPs in EC2 Classic can be imported using their Public IP, e.g.

```
$ terraform import aws_eip.bar 52.0.0.0
```

## » `aws_eip_association`

Provides an AWS EIP Association as a top level resource, to associate and disassociate Elastic IPs from AWS Instances and Network Interfaces.

**NOTE:** `aws_eip_association` is useful in scenarios where EIPs are either pre-existing or distributed to customers or users and therefore cannot be changed.

## » Example Usage

```
resource "aws_eip_association" "eip_assoc" {
  instance_id   = "${aws_instance.web.id}"
  allocation_id = "${aws_eip.example.id}"
}

resource "aws_instance" "web" {
  ami           = "ami-21f78e11"
  availability_zone = "us-west-2a"
  instance_type  = "t1.micro"

  tags {
    Name = "HelloWorld"
  }
}

resource "aws_eip" "example" {
  vpc = true
}
```

## » Argument Reference

The following arguments are supported:

- `allocation_id` - (Optional) The allocation ID. This is required for EC2-VPC.
- `allow_reassociation` - (Optional, Boolean) Whether to allow an Elastic IP to be re-associated. Defaults to `true` in VPC.
- `instance_id` - (Optional) The ID of the instance. This is required for EC2-Classic. For EC2-VPC, you can specify either the instance ID or the network interface ID, but not both. The operation fails if you specify an instance ID unless exactly one network interface is attached.

- **network\_interface\_id** - (Optional) The ID of the network interface. If the instance has more than one network interface, you must specify a network interface ID.
- **private\_ip\_address** - (Optional) The primary or secondary private IP address to associate with the Elastic IP address. If no private IP address is specified, the Elastic IP address is associated with the primary private IP address.
- **public\_ip** - (Optional) The Elastic IP address. This is required for EC2-Classic.

## » Attributes Reference

- **association\_id** - The ID that represents the association of the Elastic IP address with an instance.
- **allocation\_id** - As above
- **instance\_id** - As above
- **network\_interface\_id** - As above
- **private\_ip\_address** - As above
- **public\_ip** - As above

## » `aws_elb`

Provides an Elastic Load Balancer resource, also known as a "Classic Load Balancer" after the release of Application/Network Load Balancers.

**NOTE on ELB Instances and ELB Attachments:** Terraform currently provides both a standalone ELB Attachment resource (describing an instance attached to an ELB), and an ELB resource with **instances** defined in-line. At this time you cannot use an ELB with in-line instances in conjunction with a ELB Attachment resources. Doing so will cause a conflict and will overwrite attachments.

## » Example Usage

```
# Create a new load balancer
resource "aws_elb" "bar" {
  name                = "foobar-terraform-elb"
  availability_zones = ["us-west-2a", "us-west-2b", "us-west-2c"]

  access_logs {
    bucket      = "foo"
    bucket_prefix = "bar"
    interval    = 60
  }
}
```

```

}

listener {
  instance_port      = 8000
  instance_protocol  = "http"
  lb_port            = 80
  lb_protocol        = "http"
}

listener {
  instance_port      = 8000
  instance_protocol  = "http"
  lb_port            = 443
  lb_protocol        = "https"
  ssl_certificate_id = "arn:aws:iam::123456789012:server-certificate/certName"
}

health_check {
  healthy_threshold      = 2
  unhealthy_threshold    = 2
  timeout                = 3
  target                 = "HTTP:8000/"
  interval               = 30
}

instances              = ["${aws_instance.foo.id}"]
cross_zone_load_balancing = true
idle_timeout            = 400
connection_draining     = true
connection_draining_timeout = 400

tags {
  Name = "foobar-terraform-elb"
}
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The name of the ELB. By default generated by Terraform.
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **access\_logs** - (Optional) An Access Logs block. Access Logs documented

below.

- **availability\_zones** - (Required for an EC2-classic ELB) The AZ's to serve traffic in.
- **security\_groups** - (Optional) A list of security group IDs to assign to the ELB. Only valid if creating an ELB within a VPC
- **subnets** - (Required for a VPC ELB) A list of subnet IDs to attach to the ELB.
- **instances** - (Optional) A list of instance ids to place in the ELB pool.
- **internal** - (Optional) If true, ELB will be an internal ELB.
- **listener** - (Required) A list of listener blocks. Listeners documented below.
- **health\_check** - (Optional) A health\_check block. Health Check documented below.
- **cross\_zone\_load\_balancing** - (Optional) Enable cross-zone load balancing. Default: **true**
- **idle\_timeout** - (Optional) The time in seconds that the connection is allowed to be idle. Default: 60
- **connection\_draining** - (Optional) Boolean to enable connection draining. Default: **false**
- **connection\_draining\_timeout** - (Optional) The time in seconds to allow for connections to drain. Default: 300
- **tags** - (Optional) A mapping of tags to assign to the resource.

Exactly one of **availability\_zones** or **subnets** must be specified: this determines if the ELB exists in a VPC or in EC2-classic.

Access Logs (**access\_logs**) support the following:

- **bucket** - (Required) The S3 bucket name to store the logs in.
- **bucket\_prefix** - (Optional) The S3 bucket prefix. Logs are stored in the root if not configured.
- **interval** - (Optional) The publishing interval in minutes. Default: 60 minutes.
- **enabled** - (Optional) Boolean to enable / disable **access\_logs**. Default is **true**

Listeners (**listener**) support the following:

- **instance\_port** - (Required) The port on the instance to route to
- **instance\_protocol** - (Required) The protocol to use to the instance. Valid values are HTTP, HTTPS, TCP, or SSL
- **lb\_port** - (Required) The port to listen on for the load balancer
- **lb\_protocol** - (Required) The protocol to listen on. Valid values are HTTP, HTTPS, TCP, or SSL
- **ssl\_certificate\_id** - (Optional) The ARN of an SSL certificate you have uploaded to AWS IAM. **Note ECDSA-specific restrictions below. Only valid when lb\_protocol is either HTTPS or SSL**

Health Check (**health\_check**) supports the following:



- **healthy\_threshold** - (Required) The number of checks before the instance is declared healthy.
- **unhealthy\_threshold** - (Required) The number of checks before the instance is declared unhealthy.
- **target** - (Required) The target of the check. Valid pattern is "\${PROTOCOL}:\${PORT}\${PATH}", where PROTOCOL values are:
  - HTTP, HTTPS - PORT and PATH are required
  - TCP, SSL - PORT is required, PATH is not supported
- **interval** - (Required) The interval between checks.
- **timeout** - (Required) The length of time before the check times out.

## » Note on ECDSA Key Algorithm

If the ARN of the `ssl_certificate_id` that is pointed to references a certificate that was signed by an ECDSA key, note that ELB only supports the P256 and P384 curves. Using a certificate signed by a key using a different curve could produce the error `ERR_SSL_VERSION_OR_CIPHER_MISMATCH` in your browser.

## » Attributes Reference

The following attributes are exported:

- **id** - The name of the ELB
- **arn** - The ARN of the ELB
- **name** - The name of the ELB
- **dns\_name** - The DNS name of the ELB
- **instances** - The list of instances in the ELB
- **source\_security\_group** - The name of the security group that you can use as part of your inbound rules for your load balancer's back-end application instances. Use this for Classic or Default VPC only.
- **source\_security\_group\_id** - The ID of the security group that you can use as part of your inbound rules for your load balancer's back-end application instances. Only available on ELBs launched in a VPC.
- **zone\_id** - The canonical hosted zone ID of the ELB (to be used in a Route 53 Alias record)

## » Import

ELBs can be imported using the **name**, e.g.

```
$ terraform import aws_elb.bar elb-production-12345
```

## » aws\_\_elb\_\_attachment

Provides an Elastic Load Balancer Attachment resource.

**NOTE on ELB Instances and ELB Attachments:** Terraform currently provides both a standalone ELB Attachment resource (describing an instance attached to an ELB), and an Elastic Load Balancer resource with **instances** defined in-line. At this time you cannot use an ELB with in-line instances in conjunction with an ELB Attachment resource. Doing so will cause a conflict and will overwrite attachments.

### » Example Usage

```
# Create a new load balancer attachment
resource "aws_elb_attachment" "baz" {
  elb      = "${aws_elb.bar.id}"
  instance = "${aws_instance.foo.id}"
}
```

### » Argument Reference

The following arguments are supported:

- **elb** - (Required) The name of the ELB.
- **instance** - (Required) Instance ID to place in the ELB pool.

## » aws\_\_instance

Provides an EC2 instance resource. This allows instances to be created, updated, and deleted. Instances also support provisioning.

### » Example Usage

```
# Create a new instance of the latest Ubuntu 14.04 on an
# t2.micro node with an AWS Tag naming it "HelloWorld"
provider "aws" {
  region = "us-west-2"
}

data "aws_ami" "ubuntu" {
  most_recent = true
```

```

filter {
  name = "name"
  values = ["ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-*"]
}

filter {
  name = "virtualization-type"
  values = ["hvm"]
}

owners = ["099720109477"] # Canonical
}

resource "aws_instance" "web" {
  ami = "${data.aws_ami.ubuntu.id}"
  instance_type = "t2.micro"

  tags {
    Name = "HelloWorld"
  }
}

```

## » Argument Reference

The following arguments are supported:

- **ami** - (Required) The AMI to use for the instance.
- **availability\_zone** - (Optional) The AZ to start the instance in.
- **placement\_group** - (Optional) The Placement Group to start the instance in.
- **tenancy** - (Optional) The tenancy of the instance (if the instance is running in a VPC). An instance with a tenancy of `dedicated` runs on single-tenant hardware. The host tenancy is not supported for the `import-instance` command.
- **ebs\_optimized** - (Optional) If true, the launched EC2 instance will be EBS-optimized.
- **disable\_api\_termination** - (Optional) If true, enables EC2 Instance Termination Protection
- **instance\_initiated\_shutdown\_behavior** - (Optional) Shutdown behavior for the instance. Amazon defaults this to `stop` for EBS-backed instances and `terminate` for instance-store instances. Cannot be set on instance-store instances. See Shutdown Behavior for more information.
- **instance\_type** - (Required) The type of instance to start. Updates to this field will trigger a stop/start of the EC2 instance.
- **key\_name** - (Optional) The key name to use for the instance.

- **get\_password\_data** - (Optional) If true, wait for password data to become available and retrieve it. Useful for getting the administrator password for instances running Microsoft Windows. The password data is exported to the **password\_data** attribute. See `GetPasswordData` for more information.
- **monitoring** - (Optional) If true, the launched EC2 instance will have detailed monitoring enabled. (Available since v0.6.0)
- **security\_groups** - (Optional) A list of security group names to associate with. If you are creating Instances in a VPC, use **vpc\_security\_group\_ids** instead.
- **vpc\_security\_group\_ids** - (Optional) A list of security group IDs to associate with.
- **subnet\_id** - (Optional) The VPC Subnet ID to launch in.
- **associate\_public\_ip\_address** - (Optional) Associate a public ip address with an instance in a VPC. Boolean value.
- **private\_ip** - (Optional) Private IP address to associate with the instance in a VPC.
- **source\_dest\_check** - (Optional) Controls if traffic is routed to the instance when the destination address does not match the instance. Used for NAT or VPNs. Defaults true.
- **user\_data** - (Optional) The user data to provide when launching the instance. Do not pass gzip-compressed data via this argument; see **user\_data\_base64** instead.
- **user\_data\_base64** - (Optional) Can be used instead of **user\_data** to pass base64-encoded binary data directly. Use this instead of **user\_data** whenever the value is not a valid UTF-8 string. For example, gzip-encoded user data must be base64-encoded and passed via this argument to avoid corruption.
- **iam\_instance\_profile** - (Optional) The IAM Instance Profile to launch the instance with. Specified as the name of the Instance Profile. Ensure your credentials have the correct permission to assign the instance profile according to the EC2 documentation, notably **iam:PassRole**.
- **ipv6\_address\_count** - (Optional) A number of IPv6 addresses to associate with the primary network interface. Amazon EC2 chooses the IPv6 addresses from the range of your subnet.
- **ipv6\_addresses** - (Optional) Specify one or more IPv6 addresses from the range of the subnet to associate with the primary network interface
- **tags** - (Optional) A mapping of tags to assign to the resource.
- **volume\_tags** - (Optional) A mapping of tags to assign to the devices created by the instance at launch time.
- **root\_block\_device** - (Optional) Customize details about the root block device of the instance. See Block Devices below for details.
- **ebs\_block\_device** - (Optional) Additional EBS block devices to attach to the instance. See Block Devices below for details.
- **ephemeral\_block\_device** - (Optional) Customize Ephemeral (also known as "Instance Store") volumes on the instance. See Block Devices below

for details.

- **network\_interface** - (Optional) Customize network interfaces to be attached at instance boot time. See Network Interfaces below for more details.
- **credit\_specification** - (Optional) Customize the credit specification of the instance. See Credit Specification below for more details.

## » Timeouts

The **timeouts** block allows you to specify timeouts for certain actions:

- **create** - (Defaults to 10 mins) Used when launching the instance (until it reaches the initial **running** state)
- **update** - (Defaults to 10 mins) Used when stopping and starting the instance when necessary during update - e.g. when changing instance type
- **delete** - (Defaults to 20 mins) Used when terminating the instance

## » Block devices

Each of the **\*\_block\_device** attributes controls a portion of the AWS Instance's "Block Device Mapping". It's a good idea to familiarize yourself with AWS's Block Device Mapping docs to understand the implications of using these attributes.

The **root\_block\_device** mapping supports the following:

- **volume\_type** - (Optional) The type of volume. Can be "standard", "gp2", or "io1". (Default: "standard").
- **volume\_size** - (Optional) The size of the volume in gigabytes.
- **iops** - (Optional) The amount of provisioned IOPS. This is only valid for **volume\_type** of "io1", and must be specified if using that type
- **delete\_on\_termination** - (Optional) Whether the volume should be destroyed on instance termination (Default: **true**).

Modifying any of the **root\_block\_device** settings requires resource replacement.

Each **ebs\_block\_device** supports the following:

- **device\_name** - The name of the device to mount.
- **snapshot\_id** - (Optional) The Snapshot ID to mount.
- **volume\_type** - (Optional) The type of volume. Can be "standard", "gp2", or "io1". (Default: "standard").
- **volume\_size** - (Optional) The size of the volume in gigabytes.
- **iops** - (Optional) The amount of provisioned IOPS. This must be set with a **volume\_type** of "io1".

- `delete_on_termination` - (Optional) Whether the volume should be destroyed on instance termination (Default: `true`).
- `encrypted` - (Optional) Enables EBS encryption on the volume (Default: `false`). Cannot be used with `snapshot_id`.

Modifying any `ebs_block_device` currently requires resource replacement.

**NOTE on EBS block devices:** If you use `ebs_block_device` on an `aws_instance`, Terraform will assume management over the full set of non-root EBS block devices for the instance, and treats additional block devices as drift. For this reason, `ebs_block_device` cannot be mixed with external `aws_ebs_volume` + `aws_volume_attachment` resources for a given instance.

Each `ephemeral_block_device` supports the following:

- `device_name` - The name of the block device to mount on the instance.
- `virtual_name` - (Optional) The Instance Store Device Name (e.g. `"ephemeral0"`).
- `no_device` - (Optional) Suppresses the specified device included in the AMI's block device mapping.

Each AWS Instance type has a different set of Instance Store block devices available for attachment. AWS publishes a list of which ephemeral devices are available on each type. The devices are always identified by the `virtual_name` in the format `"ephemeral{0..N}"`.

**NOTE:** Currently, changes to `*_block_device` configuration of *existing* resources cannot be automatically detected by Terraform. After making updates to block device configuration, resource recreation can be manually triggered by using the `taint` command.

## » Network Interfaces

Each of the `network_interface` blocks attach a network interface to an EC2 Instance during boot time. However, because the network interface is attached at boot-time, replacing/modifying the network interface **WILL** trigger a recreation of the EC2 Instance. If you should need at any point to detach/modify/re-attach a network interface to the instance, use the `aws_network_interface` or `aws_network_interface_attachment` resources instead.

The `network_interface` configuration block *does*, however, allow users to supply their own network interface to be used as the default network interface on an EC2 Instance, attached at `eth0`.

Each `network_interface` block supports the following:

- `device_index` - (Required) The integer index of the network interface attachment. Limited by instance type.

- `network_interface_id` - (Required) The ID of the network interface to attach.
- `delete_on_termination` - (Optional) Whether or not to delete the network interface on instance termination. Defaults to `false`.

## » Credit Specification

Credit specification can be applied/modified to the EC2 Instance at any time.

The `credit_specification` block supports the following:

- `cpu_credits` - (Optional) The credit option for CPU usage.

## » Example

```
resource "aws_vpc" "my_vpc" {
  cidr_block = "172.16.0.0/16"
  tags {
    Name = "tf-example"
  }
}

resource "aws_subnet" "my_subnet" {
  vpc_id = "${aws_vpc.my_vpc.id}"
  cidr_block = "172.16.10.0/24"
  availability_zone = "us-west-2a"
  tags {
    Name = "tf-example"
  }
}

resource "aws_network_interface" "foo" {
  subnet_id = "${aws_subnet.my_subnet.id}"
  private_ips = ["172.16.10.100"]
  tags {
    Name = "primary_network_interface"
  }
}

resource "aws_instance" "foo" {
  ami = "ami-22b9a343" # us-west-2
  instance_type = "t2.micro"
  network_interface {
    network_interface_id = "${aws_network_interface.foo.id}"
    device_index = 0
  }
}
```

```
}  
}
```

## » Attributes Reference

The following attributes are exported:

- `id` - The instance ID.
- `availability_zone` - The availability zone of the instance.
- `placement_group` - The placement group of the instance.
- `key_name` - The key name of the instance
- `password_data` - Base-64 encoded encrypted password data for the instance. Useful for getting the administrator password for instances running Microsoft Windows. This attribute is only exported if `get_password_data` is true. Note that this encrypted value will be stored in the state file, as with all exported attributes. See `GetPasswordData` for more information.
- `public_dns` - The public DNS name assigned to the instance. For EC2-VPC, this is only available if you've enabled DNS hostnames for your VPC
- `public_ip` - The public IP address assigned to the instance, if applicable. **NOTE:** If you are using an `aws_eip` with your instance, you should refer to the EIP's address directly and not use `public_ip`, as this field will change after the EIP is attached.
- `ipv6_addresses` - A list of assigned IPv6 addresses, if any
- `network_interface_id` - The ID of the network interface that was created with the instance.
- `primary_network_interface_id` - The ID of the instance's primary network interface.
- `private_dns` - The private DNS name assigned to the instance. Can only be used inside the Amazon EC2, and only available if you've enabled DNS hostnames for your VPC
- `private_ip` - The private IP address assigned to the instance
- `security_groups` - The associated security groups.
- `vpc_security_group_ids` - The associated security groups in non-default VPC
- `subnet_id` - The VPC subnet ID.
- `credit_specification` - Credit specification of instance.

For any `root_block_device` and `ebs_block_device` the `volume_id` is exported. e.g. `aws_instance.web.root_block_device.0.volume_id`

## » Import

Instances can be imported using the `id`, e.g.



```
$ terraform import aws_instance.web i-12345678
```

## » **aws\_\_key\_\_pair**

Provides an EC2 key pair resource. A key pair is used to control login access to EC2 instances.

Currently this resource requires an existing user-supplied key pair. This key pair's public key will be registered with AWS to allow logging-in to EC2 instances.

When importing an existing key pair the public key material may be in any format supported by AWS. Supported formats (per the AWS documentation) are:

- OpenSSH public key format (the format in `~/.ssh/authorized_keys`)
- Base64 encoded DER format
- SSH public key file format as specified in RFC4716

## » **Example Usage**

```
resource "aws_key_pair" "deployer" {  
  key_name    = "deployer-key"  
  public_key = "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQD3F6tyPEFEzVOLX3X8BsXdMsQz1x2cEikKDEY"  
}
```

## » **Argument Reference**

The following arguments are supported:

- **key\_name** - (Optional) The name for the key pair.
- **key\_name\_prefix** - (Optional) Creates a unique name beginning with the specified prefix. Conflicts with **key\_name**.
- **public\_key** - (Required) The public key material.

## » **Attributes Reference**

The following attributes are exported:

- **key\_name** - The key pair name.
- **fingerprint** - The MD5 public key fingerprint as specified in section 4 of RFC 4716.

## » Import

Key Pairs can be imported using the `key_name`, e.g.

```
$ terraform import aws_key_pair.deployer deployer-key
```

## » `aws_launch_configuration`

Provides a resource to create a new launch configuration, used for autoscaling groups.

## » Example Usage

```
data "aws_ami" "ubuntu" {
  most_recent = true

  filter {
    name     = "name"
    values   = ["ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-*"]
  }

  filter {
    name     = "virtualization-type"
    values   = ["hvm"]
  }

  owners = ["099720109477"] # Canonical
}

resource "aws_launch_configuration" "as_conf" {
  name          = "web_config"
  image_id      = "${data.aws_ami.ubuntu.id}"
  instance_type = "t2.micro"
}
```

## » Using with AutoScaling Groups

Launch Configurations cannot be updated after creation with the Amazon Web Service API. In order to update a Launch Configuration, Terraform will destroy the existing resource and create a replacement. In order to effectively use a Launch Configuration resource with an AutoScaling Group resource, it's recommended to specify `create_before_destroy` in a lifecycle block. Either

omit the Launch Configuration `name` attribute, or specify a partial name with `name_prefix`. Example:

```
data "aws_ami" "ubuntu" {
  most_recent = true

  filter {
    name     = "name"
    values   = ["ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-*"]
  }

  filter {
    name     = "virtualization-type"
    values   = ["hvm"]
  }

  owners = ["099720109477"] # Canonical
}

resource "aws_launch_configuration" "as_conf" {
  name_prefix   = "terraform-lc-example-"
  image_id      = "${data.aws_ami.ubuntu.id}"
  instance_type = "t2.micro"

  lifecycle {
    create_before_destroy = true
  }
}

resource "aws_autoscaling_group" "bar" {
  name                  = "terraform-asg-example"
  launch_configuration = "${aws_launch_configuration.as_conf.name}"
  min_size              = 1
  max_size              = 2

  lifecycle {
    create_before_destroy = true
  }
}
```

With this setup Terraform generates a unique name for your Launch Configuration and can then update the AutoScaling Group without conflict before destroying the previous Launch Configuration.

## » Using with Spot Instances

Launch configurations can set the spot instance pricing to be used for the Auto Scaling Group to reserve instances. Simply specifying the `spot_price` parameter will set the price on the Launch Configuration which will attempt to reserve your instances at this price. See the AWS Spot Instance documentation for more information or how to launch Spot Instances with Terraform.

```
data "aws_ami" "ubuntu" {
  most_recent = true

  filter {
    name     = "name"
    values   = ["ubuntu/images/hvm-ssd/ubuntu-trusty-14.04-amd64-server-*"]
  }

  filter {
    name     = "virtualization-type"
    values   = ["hvm"]
  }

  owners = ["099720109477"] # Canonical
}

resource "aws_launch_configuration" "as_conf" {
  image_id        = "${data.aws_ami.ubuntu.id}"
  instance_type   = "m4.large"
  spot_price      = "0.001"

  lifecycle {
    create_before_destroy = true
  }
}

resource "aws_autoscaling_group" "bar" {
  name                  = "terraform-asg-example"
  launch_configuration = "${aws_launch_configuration.as_conf.name}"
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Optional) The name of the launch configuration. If you leave this blank, Terraform will auto-generate a unique name.

- **name\_prefix** - (Optional) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **image\_id** - (Required) The EC2 image ID to launch.
- **instance\_type** - (Required) The size of instance to launch.
- **iam\_instance\_profile** - (Optional) The IAM instance profile to associate with launched instances.
- **key\_name** - (Optional) The key name that should be used for the instance.
- **security\_groups** - (Optional) A list of associated security group IDs.
- **associate\_public\_ip\_address** - (Optional) Associate a public ip address with an instance in a VPC.
- **vpc\_classic\_link\_id** - (Optional) The ID of a ClassicLink-enabled VPC. Only applies to EC2-Classic instances. (eg. **vpc-2730681a**)
- **vpc\_classic\_link\_security\_groups** - (Optional) The IDs of one or more security groups for the specified ClassicLink-enabled VPC (eg. **sg-46ae3d11**).
- **user\_data** - (Optional) The user data to provide when launching the instance. Do not pass gzip-compressed data via this argument; see **user\_data\_base64** instead.
- **user\_data\_base64** - (Optional) Can be used instead of **user\_data** to pass base64-encoded binary data directly. Use this instead of **user\_data** whenever the value is not a valid UTF-8 string. For example, gzip-encoded user data must be base64-encoded and passed via this argument to avoid corruption.
- **enable\_monitoring** - (Optional) Enables/disables detailed monitoring. This is enabled by default.
- **ebs\_optimized** - (Optional) If true, the launched EC2 instance will be EBS-optimized.
- **root\_block\_device** - (Optional) Customize details about the root block device of the instance. See Block Devices below for details.
- **ebs\_block\_device** - (Optional) Additional EBS block devices to attach to the instance. See Block Devices below for details.
- **ephemeral\_block\_device** - (Optional) Customize Ephemeral (also known as "Instance Store") volumes on the instance. See Block Devices below for details.
- **spot\_price** - (Optional) The price to use for reserving spot instances.
- **placement\_tenancy** - (Optional) The tenancy of the instance. Valid values are "default" or "dedicated", see AWS's Create Launch Configuration for more details

## » Block devices

Each of the **\*\_block\_device** attributes controls a portion of the AWS Launch Configuration's "Block Device Mapping". It's a good idea to familiarize yourself with AWS's Block Device Mapping docs to understand the implications of using these attributes.

The `root_block_device` mapping supports the following:

- `volume_type` - (Optional) The type of volume. Can be `"standard"`, `"gp2"`, or `"io1"`. (Default: `"standard"`).
- `volume_size` - (Optional) The size of the volume in gigabytes.
- `iops` - (Optional) The amount of provisioned IOPS. This must be set with a `volume_type` of `"io1"`.
- `delete_on_termination` - (Optional) Whether the volume should be destroyed on instance termination (Default: `true`).

Modifying any of the `root_block_device` settings requires resource replacement.

Each `ebs_block_device` supports the following:

- `device_name` - (Required) The name of the device to mount.
- `snapshot_id` - (Optional) The Snapshot ID to mount.
- `volume_type` - (Optional) The type of volume. Can be `"standard"`, `"gp2"`, or `"io1"`. (Default: `"standard"`).
- `volume_size` - (Optional) The size of the volume in gigabytes.
- `iops` - (Optional) The amount of provisioned IOPS. This must be set with a `volume_type` of `"io1"`.
- `delete_on_termination` - (Optional) Whether the volume should be destroyed on instance termination (Default: `true`).
- `encrypted` - (Optional) Whether the volume should be encrypted or not. Do not use this option if you are using `snapshot_id` as the encrypted flag will be determined by the snapshot. (Default: `false`).

Modifying any `ebs_block_device` currently requires resource replacement.

Each `ephemeral_block_device` supports the following:

- `device_name` - The name of the block device to mount on the instance.
- `virtual_name` - The Instance Store Device Name (e.g. `"ephemeral0"`)

Each AWS Instance type has a different set of Instance Store block devices available for attachment. AWS publishes a list of which ephemeral devices are available on each type. The devices are always identified by the `virtual_name` in the format `"ephemeral{0..N}"`.

**NOTE:** Changes to `*_block_device` configuration of *existing* resources cannot currently be detected by Terraform. After updating to block device configuration, resource recreation can be manually triggered by using the `taint` command.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the launch configuration.

- `name` - The name of the launch configuration.

## » Import

Launch configurations can be imported using the `name`, e.g.

```
$ terraform import aws_launch_configuration.as_conf terraform-lg-123456
```

## » `aws_launch_template`

Provides an EC2 launch template resource. Can be used to create instances or auto scaling groups.

## » Example Usage

```
resource "aws_launch_template" "foo" {
  name = "foo"

  block_device_mappings {
    device_name = "/dev/sda1"
    ebs {
      volume_size = 20
    }
  }

  credit_specification {
    cpu_credits = "standard"
  }

  disable_api_termination = true

  ebs_optimized = true

  elastic_gpu_specifications {
    type = "test"
  }

  iam_instance_profile {
    name = "test"
  }

  image_id = "ami-test"
```

```

instance_initiated_shutdown_behavior = "terminate"

instance_market_options {
  market_type = "spot"
}

instance_type = "t2.micro"

kernel_id = "test"

key_name = "test"

monitoring {
  enabled = true
}

network_interfaces {
  associate_public_ip_address = true
}

placement {
  availability_zone = "us-west-2a"
}

ram_disk_id = "test"

vpc_security_group_ids = ["sg-12345678"]

tag_specifications {
  resource_type = "instance"
  tags {
    Name = "test"
  }
}
}

```

## » Argument Reference

The following arguments are supported:

- **name** - The name of the launch template. If you leave this blank, Terraform will auto-generate a unique name.
- **name\_prefix** - Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **description** - Description of the launch template.



- **block\_device\_mappings** - Specify volumes to attach to the instance besides the volumes specified by the AMI. See Block Devices below for details.
- **credit\_specification** - Customize the credit specification of the instance. See Credit Specification below for more details.
- **disable\_api\_termination** - If **true**, enables EC2 Instance Termination Protection
- **ebs\_optimized** - If **true**, the launched EC2 instance will be EBS-optimized.
- **elastic\_gpu\_specifications** - The elastic GPU to attach to the instance. See Elastic GPU below for more details.
- **iam\_instance\_profile** - The IAM Instance Profile to launch the instance with. See Instance Profile below for more details.
- **image\_id** - The AMI from which to launch the instance.
- **instance\_initiated\_shutdown\_behavior** - Shutdown behavior for the instance. Can be **stop** or **terminate**. (Default: **stop**).
- **instance\_market\_options** - The market (purchasing) option for the instance. See Market Options below for details.
- **instance\_type** - The type of the instance.
- **kernel\_id** - The kernel ID.
- **key\_name** - The key name to use for the instance.
- **monitoring** - The monitoring option for the instance. See Monitoring below for more details.
- **network\_interfaces** - Customize network interfaces to be attached at instance boot time. See Network Interfaces below for more details.
- **placement** - The placement of the instance. See Placement below for more details.
- **ram\_disk\_id** - The ID of the RAM disk.
- **security\_group\_names** - A list of security group names to associate with. If you are creating Instances in a VPC, use **vpc\_security\_group\_ids** instead.
- **vpc\_security\_group\_ids** - A list of security group IDs to associate with.
- **tag\_specifications** - The tags to apply to the resources during launch. See Tags below for more details.
- **user\_data** - The user data to provide when launching the instance.

## » Block devices

Configure additional volumes of the instance besides specified by the AMI. It's a good idea to familiarize yourself with AWS's Block Device Mapping docs to understand the implications of using these attributes.

To find out more information for an existing AMI to override the configuration, such as **device\_name**, you can use the AWS CLI `ec2 describe-images` command.

Each **block\_device\_mappings** supports the following:

- `device_name` - The name of the device to mount.
- `ebs` - Configure EBS volume properties.
- `no_device` - Suppresses the specified device included in the AMI's block device mapping.
- `virtual_name` - The Instance Store Device Name (e.g. "ephemeral0").

The `ebs` block supports the following:

- `delete_on_termination` - Whether the volume should be destroyed on instance termination (Default: `true`).
- `encrypted` - Enables EBS encryption on the volume (Default: `false`). Cannot be used with `snapshot_id`.
- `iops` - The amount of provisioned IOPS. This must be set with a `volume_type` of "io1".
- `kms_key_id` - AWS Key Management Service (AWS KMS) customer master key (CMK) to use when creating the encrypted volume. `encrypted` must be set to `true` when this is set.
- `snapshot_id` - The Snapshot ID to mount.
- `volume_size` - The size of the volume in gigabytes.
- `volume_type` - The type of volume. Can be "standard", "gp2", or "io1". (Default: "standard").

## » Credit Specification

Credit specification can be applied/modified to the EC2 Instance at any time.

The `credit_specification` block supports the following:

- `cpu_credits` - The credit option for CPU usage. Can be "standard" or "unlimited". (Default: "standard").

## » Elastic GPU

Attach an elastic GPU the instance.

The `elastic_gpu_specifications` block supports the following:

- `type` - The Elastic GPU Type

## » Instance Profile

The IAM Instance Profile to attach.

The `iam_instance_profile` block supports the following:

- `arn` - The Amazon Resource Name (ARN) of the instance profile.
- `name` - The name of the instance profile.

## » Market Options

The market (purchasing) option for the instances.

The `instance_market_options` block supports the following:

- `market_type` - The market type. Can be `spot`.
- `spot_options` - The options for Spot Instance

The `spot_options` block supports the following:

- `block_duration_minutes` - The required duration in minutes. This value must be a multiple of 60.
- `instance_interruption_behavior` - The behavior when a Spot Instance is interrupted. Can be `hibernate`, `stop`, or `terminate`. (Default: `terminate`).
- `max_price` - The maximum hourly price you're willing to pay for the Spot Instances.
- `spot_instance_type` - The Spot Instance request type. Can be `one-time`, or `persistent`.
- `valid_until` - The end date of the request.

## » Monitoring

The `monitoring` block supports the following:

- `enabled` - If `true`, the launched EC2 instance will have detailed monitoring enabled.

## » Network Interfaces

Attaches one or more Network Interfaces to the instance.

Each `network_interfaces` block supports the following:

- `associate_public_ip_address` - Associate a public ip address with the network interface. Boolean value.
- `delete_on_termination` - Whether the network interface should be destroyed on instance termination.
- `description` - Description of the network interface.
- `device_index` - The integer index of the network interface attachment.
- `ipv6_addresses` - One or more specific IPv6 addresses from the IPv6 CIDR block range of your subnet.
- `network_interface_id` - The ID of the network interface to attach.
- `private_ip_address` - The primary private IPv4 address.
- `ipv4_addresses` - One or more private IPv4 addresses to associate.
- `security_groups` - A list of security group IDs to associate.
- `subnet_id` - The VPC Subnet ID to associate.

## » Placement

The Placement Group of the instance.

The `placement` block supports the following:

- `affinity` - The affinity setting for an instance on a Dedicated Host.
- `availability_zone` - The Availability Zone for the instance.
- `group_name` - The name of the placement group for the instance.
- `host_id` - The ID of the Dedicated Host for the instance.
- `spread_domain` - Reserved for future use.
- `tenancy` - The tenancy of the instance (if the instance is running in a VPC). Can be `default`, `dedicated`, or `host`.

## » Tags

The tags to apply to the resources during launch. You can tag instances and volumes.

Each `tag_specifications` block supports the following:

- `resource_type` - The type of resource to tag.
- `tags` - A mapping of tags to assign to the resource.

## » Attributes Reference

The following attributes are exported along with all argument references:

- `id` - The ID of the launch template.
- `default_version` - The default version of the launch template.
- `latest_version` - The latest version of the launch template.

## » Import

Launch Templates can be imported using the `id`, e.g.

```
$ terraform import aws_launch_template.web lt-12345678
```

## » `aws_lb_cookie_stickiness_policy`

Provides a load balancer cookie stickiness policy, which allows an ELB to control the sticky session lifetime of the browser.

## » Example Usage

```
resource "aws_elb" "lb" {
  name                = "test-lb"
  availability_zones = ["us-east-1a"]

  listener {
    instance_port      = 8000
    instance_protocol = "http"
    lb_port            = 80
    lb_protocol        = "http"
  }
}

resource "aws_lb_cookie_stickiness_policy" "foo" {
  name                = "foo-policy"
  load_balancer       = "${aws_elb.lb.id}"
  lb_port             = 80
  cookie_expiration_period = 600
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the stickiness policy.
- **load\_balancer** - (Required) The load balancer to which the policy should be attached.
- **lb\_port** - (Required) The load balancer port to which the policy should be applied. This must be an active listener on the load balancer.
- **cookie\_expiration\_period** - (Optional) The time period after which the session cookie should be considered stale, expressed in seconds.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the policy.
- **name** - The name of the stickiness policy.
- **load\_balancer** - The load balancer to which the policy is attached.
- **lb\_port** - The load balancer port to which the policy is applied.
- **cookie\_expiration\_period** - The time period after which the session cookie is considered stale, expressed in seconds.

## » `aws_lb_ssl_negotiation_policy`

Provides a load balancer SSL negotiation policy, which allows an ELB to control the ciphers and protocols that are supported during SSL negotiations between a client and a load balancer.

### » Example Usage

```
resource "aws_elb" "lb" {
  name                = "test-lb"
  availability_zones = ["us-east-1a"]

  listener {
    instance_port      = 8000
    instance_protocol  = "https"
    lb_port            = 443
    lb_protocol        = "https"
    ssl_certificate_id = "arn:aws:iam::123456789012:server-certificate/certName"
  }
}

resource "aws_lb_ssl_negotiation_policy" "foo" {
  name                = "foo-policy"
  load_balancer       = "${aws_elb.lb.id}"
  lb_port             = 443

  attribute {
    name = "Protocol-TLSv1"
    value = "false"
  }

  attribute {
    name = "Protocol-TLSv1.1"
    value = "false"
  }

  attribute {
    name = "Protocol-TLSv1.2"
    value = "true"
  }

  attribute {
    name = "Server-Defined-Cipher-Order"
    value = "true"
  }
}
```

```

    }

    attribute {
        name = "ECDHE-RSA-AES128-GCM-SHA256"
        value = "true"
    }

    attribute {
        name = "AES128-GCM-SHA256"
        value = "true"
    }

    attribute {
        name = "EDH-RSA-DES-CBC3-SHA"
        value = "false"
    }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the SSL negotiation policy.
- **load\_balancer** - (Required) The load balancer to which the policy should be attached.
- **lb\_port** - (Required) The load balancer port to which the policy should be applied. This must be an active listener on the load balancer.
- **attribute** - (Optional) An SSL Negotiation policy attribute. Each has two properties:
  - **name** - The name of the attribute
  - **value** - The value of the attribute

To set your attributes, please see the AWS Elastic Load Balancing Developer Guide for a listing of the supported SSL protocols, SSL options, and SSL ciphers.

**NOTE:** The AWS documentation references Server Order Preference, which the AWS Elastic Load Balancing API refers to as **Server-Defined-Cipher-Order**. If you wish to set Server Order Preference, use this value instead.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the policy.
- **name** - The name of the stickiness policy.
- **load\_balancer** - The load balancer to which the policy is attached.

- `lb_port` - The load balancer port to which the policy is applied.
- `attribute` - The SSL Negotiation policy attributes.

## » `aws_elb_load_balancer_backend_server_policy`

Attaches a load balancer policy to an ELB backend server.

### » Example Usage

```
resource "aws_elb" "wu-tang" {
  name                = "wu-tang"
  availability_zones = ["us-east-1a"]

  listener {
    instance_port      = 443
    instance_protocol  = "http"
    lb_port            = 443
    lb_protocol        = "https"
    ssl_certificate_id = "arn:aws:iam::000000000000:server-certificate/wu-tang.net"
  }

  tags {
    Name = "wu-tang"
  }
}

resource "aws_load_balancer_policy" "wu-tang-ca-pubkey-policy" {
  load_balancer_name = "${aws_elb.wu-tang.name}"
  policy_name        = "wu-tang-ca-pubkey-policy"
  policy_type_name   = "PublicKeyPolicyType"

  policy_attribute = {
    name = "PublicKey"
    value = "${file("wu-tang-pubkey")}"
  }
}

resource "aws_load_balancer_policy" "wu-tang-root-ca-backend-auth-policy" {
  load_balancer_name = "${aws_elb.wu-tang.name}"
  policy_name        = "wu-tang-root-ca-backend-auth-policy"
  policy_type_name   = "BackendServerAuthenticationPolicyType"

  policy_attribute = {
```



```

        name = "PublicKeyPolicyName"
        value = "${aws_load_balancer_policy.wu-tang-root-ca-pubkey-policy.policy_name}"
    }
}

resource "aws_load_balancer_backend_server_policy" "wu-tang-backend-auth-policies-443" {
    load_balancer_name = "${aws_elb.wu-tang.name}"
    instance_port      = 443

    policy_names = [
        "${aws_load_balancer_policy.wu-tang-root-ca-backend-auth-policy.policy_name}",
    ]
}

```

Where the file `pubkey` in the current directory contains only the *public key* of the certificate.

```
cat wu-tang-ca.pem | openssl x509 -pubkey -noout | grep -v '\-\-\-\-' | tr -d '\n' > wu-tang
```

This example shows how to enable backend authentication for an ELB as well as customize the TLS settings.

## » Argument Reference

The following arguments are supported:

- `load_balancer_name` - (Required) The load balancer to attach the policy to.
- `policy_names` - (Required) List of Policy Names to apply to the backend server.
- `instance_port` - (Required) The instance port to apply the policy to.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the policy.
- `load_balancer_name` - The load balancer on which the policy is defined.
- `instance_port` - The backend port the policies are applied to

## » `aws_elb_load_balancer_listener_policy`

Attaches a load balancer policy to an ELB Listener.

## » Example Usage

```
resource "aws_elb" "wu-tang" {
  name                = "wu-tang"
  availability_zones = ["us-east-1a"]

  listener {
    instance_port      = 443
    instance_protocol  = "http"
    lb_port            = 443
    lb_protocol        = "https"
    ssl_certificate_id = "arn:aws:iam::000000000000:server-certificate/wu-tang.net"
  }

  tags {
    Name = "wu-tang"
  }
}

resource "aws_load_balancer_policy" "wu-tang-ssl" {
  load_balancer_name = "${aws_elb.wu-tang.name}"
  policy_name        = "wu-tang-ssl"
  policy_type_name   = "SSLNegotiationPolicyType"

  policy_attribute = {
    name = "ECDHE-ECDSA-AES128-GCM-SHA256"
    value = "true"
  }

  policy_attribute = {
    name = "Protocol-TLSv1.2"
    value = "true"
  }
}

resource "aws_load_balancer_listener_policy" "wu-tang-listener-policies-443" {
  load_balancer_name = "${aws_elb.wu-tang.name}"
  load_balancer_port = 443

  policy_names = [
    "${aws_load_balancer_policy.wu-tang-ssl.policy_name}",
  ]
}
```

This example shows how to customize the TLS settings of an HTTPS listener.

## » Argument Reference

The following arguments are supported:

- `load_balancer_name` - (Required) The load balancer to attach the policy to.
- `load_balancer_port` - (Required) The load balancer listener port to apply the policy to.
- `policy_names` - (Required) List of Policy Names to apply to the backend server.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the policy.
- `load_balancer_name` - The load balancer on which the policy is defined.
- `load_balancer_port` - The load balancer listener port the policies are applied to

## » `aws_elb_load_balancer_policy`

Provides a load balancer policy, which can be attached to an ELB listener or backend server.

## » Example Usage

```
resource "aws_elb" "wu-tang" {
  name                = "wu-tang"
  availability_zones = ["us-east-1a"]

  listener {
    instance_port      = 443
    instance_protocol  = "http"
    lb_port            = 443
    lb_protocol        = "https"
    ssl_certificate_id = "arn:aws:iam::000000000000:server-certificate/wu-tang.net"
  }

  tags {
    Name = "wu-tang"
  }
}
```

```

resource "aws_load_balancer_policy" "wu-tang-ca-pubkey-policy" {
  load_balancer_name = "${aws_elb.wu-tang.name}"
  policy_name        = "wu-tang-ca-pubkey-policy"
  policy_type_name    = "PublicKeyPolicyType"

  policy_attribute = {
    name = "PublicKey"
    value = "${file("wu-tang-pubkey")}"
  }
}

resource "aws_load_balancer_policy" "wu-tang-root-ca-backend-auth-policy" {
  load_balancer_name = "${aws_elb.wu-tang.name}"
  policy_name        = "wu-tang-root-ca-backend-auth-policy"
  policy_type_name    = "BackendServerAuthenticationPolicyType"

  policy_attribute = {
    name = "PublicKeyPolicyName"
    value = "${aws_load_balancer_policy.wu-tang-root-ca-pubkey-policy.policy_name}"
  }
}

resource "aws_load_balancer_policy" "wu-tang-ssl" {
  load_balancer_name = "${aws_elb.wu-tang.name}"
  policy_name        = "wu-tang-ssl"
  policy_type_name    = "SSLNegotiationPolicyType"

  policy_attribute = {
    name = "ECDHE-ECDSA-AES128-GCM-SHA256"
    value = "true"
  }

  policy_attribute = {
    name = "Protocol-TLSv1.2"
    value = "true"
  }
}

resource "aws_load_balancer_backend_server_policy" "wu-tang-backend-auth-policies-443" {
  load_balancer_name = "${aws_elb.wu-tang.name}"
  instance_port      = 443

  policy_names = [
    "${aws_load_balancer_policy.wu-tang-root-ca-backend-auth-policy.policy_name}",
  ]
}

```

```

}

resource "aws_load_balancer_listener_policy" "wu-tang-listener-policies-443" {
  load_balancer_name = "${aws_elb.wu-tang.name}"
  load_balancer_port = 443

  policy_names = [
    "${aws_load_balancer_policy.wu-tang-ssl.policy_name}",
  ]
}

```

Where the file `pubkey` in the current directory contains only the *public key* of the certificate.

```
cat wu-tang-ca.pem | openssl x509 -pubkey -noout | grep -v '\-\-\-\-' | tr -d '\n' > wu-tang
```

This example shows how to enable backend authentication for an ELB as well as customize the TLS settings.

## » Argument Reference

The following arguments are supported:

- `load_balancer_name` - (Required) The load balancer on which the policy is defined.
- `policy_name` - (Required) The name of the load balancer policy.
- `policy_type_name` - (Required) The policy type.
- `policy_attribute` - (Optional) Policy attribute to apply to the policy.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the policy.
- `policy_name` - The name of the stickiness policy.
- `policy_type_name` - The policy type of the policy.
- `load_balancer_name` - The load balancer on which the policy is defined.

## » `aws__placement__group`

Provides an EC2 placement group. Read more about placement groups in AWS Docs.

## » Example Usage

```
resource "aws_placement_group" "web" {  
  name      = "hunky-dory-pg"  
  strategy = "cluster"  
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the placement group.
- **strategy** - (Required) The placement strategy.

## » Attributes Reference

The following attributes are exported:

- **id** - The name of the placement group.

## » Import

Placement groups can be imported using the **name**, e.g.

```
$ terraform import aws_placement_group.prod_pg production-placement-group
```

## » `aws__proxy__protocol__policy`

Provides a proxy protocol policy, which allows an ELB to carry a client connection information to a backend.

## » Example Usage

```
resource "aws_elb" "lb" {  
  name                = "test-lb"  
  availability_zones = ["us-east-1a"]  
  
  listener {  
    instance_port      = 25  
    instance_protocol = "tcp"  
    lb_port            = 25  
    lb_protocol        = "tcp"  
  }  
}
```

```

    }

    listener {
      instance_port      = 587
      instance_protocol = "tcp"
      lb_port            = 587
      lb_protocol        = "tcp"
    }
  }

  resource "aws_proxy_protocol_policy" "smtp" {
    load_balancer = "${aws_elb.lb.name}"
    instance_ports = ["25", "587"]
  }

```

## » Argument Reference

The following arguments are supported:

- **load\_balancer** - (Required) The load balancer to which the policy should be attached.
- **instance\_ports** - (Required) List of instance ports to which the policy should be applied. This can be specified if the protocol is SSL or TCP.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the policy.
- **load\_balancer** - The load balancer to which the policy is attached.

## » `aws_spot_datafeed_subscription`

**Note:** There is only a single subscription allowed per account.

To help you understand the charges for your Spot instances, Amazon EC2 provides a data feed that describes your Spot instance usage and pricing. This data feed is sent to an Amazon S3 bucket that you specify when you subscribe to the data feed.

## » Example Usage

```

resource "aws_s3_bucket" "default" {

```

```

    bucket = "tf-spot-datafeed"
  }

resource "aws_spot_datafeed_subscription" "default" {
  bucket = "${aws_s3_bucket.default.bucket}"
  prefix = "my_subdirectory"
}

```

## » Argument Reference

- **bucket** - (Required) The Amazon S3 bucket in which to store the Spot instance data feed.
- **prefix** - (Optional) Path of folder inside bucket to place spot pricing data.

## » Import

A Spot Datafeed Subscription can be imported using the word `spot-datafeed-subscription`, e.g.

```
$ terraform import aws_spot_datafeed_subscription.mysubscription spot-datafeed-subscription
```

## » `aws__spot__fleet__request`

Provides an EC2 Spot Fleet Request resource. This allows a fleet of Spot instances to be requested on the Spot market.

## » Example Usage

```

# Request a Spot fleet
resource "aws_spot_fleet_request" "cheap_compute" {
  iam_fleet_role      = "arn:aws:iam::12345678:role/spot-fleet"
  spot_price           = "0.03"
  allocation_strategy = "diversified"
  target_capacity      = 6
  valid_until          = "2019-11-04T20:44:20Z"

  launch_specification {
    instance_type = "m4.10xlarge"
    ami           = "ami-1234"
    spot_price    = "2.793"
    placement_tenancy = "dedicated"
  }
}

```



```

launch_specification {
  instance_type     = "m4.4xlarge"
  ami               = "ami-5678"
  key_name          = "my-key"
  spot_price        = "1.117"
  availability_zone  = "us-west-1a"
  subnet_id         = "subnet-1234"
  weighted_capacity = 35

  root_block_device {
    volume_size = "300"
    volume_type = "gp2"
  }

  tags {
    Name = "spot-fleet-example"
  }
}

```

**NOTE:** Terraform does not support the functionality where multiple `subnet_id` or `availability_zone` parameters can be specified in the same launch configuration block. If you want to specify multiple values, then separate launch configuration blocks should be used:

```

resource "aws_spot_fleet_request" "foo" {
  iam_fleet_role = "arn:aws:iam::12345678:role/spot-fleet"
  spot_price     = "0.005"
  target_capacity = 2
  valid_until    = "2019-11-04T20:44:20Z"

  launch_specification {
    instance_type = "m1.small"
    ami           = "ami-d06a90b0"
    key_name      = "my-key"
    availability_zone = "us-west-2a"
  }

  launch_specification {
    instance_type = "m3.large"
    ami           = "ami-d06a90b0"
    key_name      = "my-key"
    availability_zone = "us-west-2a"
  }

  depends_on = ["aws_iam_policy_attachment.test-attach"]
}

```

}

## » Argument Reference

Most of these arguments directly correspond to the official API.

- **iam\_fleet\_role** - (Required) Grants the Spot fleet permission to terminate Spot instances on your behalf when you cancel its Spot fleet request using `CancelSpotFleetRequests` or when the Spot fleet request expires, if you set `terminateInstancesWithExpiration`.
- **replace\_unhealthy\_instances** - (Optional) Indicates whether Spot fleet should replace unhealthy instances. Default **false**.
- **launch\_specification** - Used to define the launch configuration of the spot-fleet request. Can be specified multiple times to define different bids across different markets and instance types.

**Note:** This takes in similar but not identical inputs as **aws\_instance**. There are limitations on what you can specify. See the list of officially supported inputs in the reference documentation. Any normal **aws\_instance** parameter that corresponds to those inputs may be used.

- **spot\_price** - (Required) The bid price per unit hour.
- **wait\_for\_fulfillment** - (Optional; Default: false) If set, Terraform will wait for the Spot Request to be fulfilled, and will throw an error if the timeout of 10m is reached.
- **target\_capacity** - The number of units to request. You can choose to set the target capacity in terms of instances or a performance characteristic that is important to your application workload, such as vCPUs, memory, or I/O.
- **allocation\_strategy** - Indicates how to allocate the target capacity across the Spot pools specified by the Spot fleet request. The default is `lowestPrice`.
- **excess\_capacity\_termination\_policy** - Indicates whether running Spot instances should be terminated if the target capacity of the Spot fleet request is decreased below the current size of the Spot fleet.
- **terminate\_instances\_with\_expiration** - Indicates whether running Spot instances should be terminated when the Spot fleet request expires.
- **instance\_interruption\_behavior** - (Optional) Indicates whether a Spot instance stops or terminates when it is interrupted. Default is **terminate**.
- **valid\_until** - The end date and time of the request, in UTC ISO8601 format (for example, YYYY-MM-DDTHH:MM:SSZ). At this point, no

new Spot instance requests are placed or enabled to fulfill the request. Defaults to 24 hours.

- **load\_balancers** (Optional) A list of elastic load balancer names to add to the Spot fleet.
- **target\_group\_arns** (Optional) A list of **aws\_alb\_target\_group** ARNs, for use with Application Load Balancing.

## » Timeouts

The **timeouts** block allows you to specify timeouts for certain actions:

- **create** - (Defaults to 10 mins) Used when requesting the spot instance (only valid if **wait\_for\_fulfillment** = **true**)
- **delete** - (Defaults to 5 mins) Used when destroying the spot instance

## » Attributes Reference

The following attributes are exported:

- **id** - The Spot fleet request ID
- **spot\_request\_state** - The state of the Spot fleet request.

## » **aws\_spot\_instance\_request**

Provides an EC2 Spot Instance Request resource. This allows instances to be requested on the spot market.

Terraform always creates Spot Instance Requests with a **persistent** type, which means that for the duration of their lifetime, AWS will launch an instance with the configured details if and when the spot market will accept the requested price.

On destruction, Terraform will make an attempt to terminate the associated Spot Instance if there is one present.

**NOTE:** Because their behavior depends on the live status of the spot market, Spot Instance Requests have a unique lifecycle that makes them behave differently than other Terraform resources. Most importantly: there is **no guarantee** that a Spot Instance exists to fulfill the request at any given point in time. See the AWS Spot Instance documentation for more information.

## » Example Usage

```
# Request a spot instance at $0.03
resource "aws_spot_instance_request" "cheap_worker" {
  ami          = "ami-1234"
  spot_price   = "0.03"
  instance_type = "c4.xlarge"

  tags {
    Name = "CheapWorker"
  }
}
```

## » Argument Reference

Spot Instance Requests support all the same arguments as `aws_instance`, with the addition of:

- `spot_price` - (Required) The price to request on the spot market.
- `wait_for_fulfillment` - (Optional; Default: `false`) If set, Terraform will wait for the Spot Request to be fulfilled, and will throw an error if the timeout of 10m is reached.
- `spot_type` - (Optional; Default: `"persistent"`) If set to `"one-time"`, after the instance is terminated, the spot request will be closed. Also, Terraform can't manage one-time spot requests, just launch them.
- `launch_group` - (Optional) A launch group is a group of spot instances that launch together and terminate together. If left empty instances are launched and terminated individually.
- `block_duration_minutes` - (Optional) The required duration for the Spot instances, in minutes. This value must be a multiple of 60 (60, 120, 180, 240, 300, or 360). The duration period starts as soon as your Spot instance receives its instance ID. At the end of the duration period, Amazon EC2 marks the Spot instance for termination and provides a Spot instance termination notice, which gives the instance a two-minute warning before it terminates. Note that you can't specify an Availability Zone group or a launch group if you specify a duration.
- `instance_interruption_behaviour` - (Optional) Indicates whether a Spot instance stops or terminates when it is interrupted. Default is `terminate` as this is the current AWS behaviour.
- `valid_until` - (Optional) The end date and time of the request, in UTC RFC3339 format (for example, `YYYY-MM-DDTHH:MM:SSZ`). At this point, no new Spot instance requests are placed or enabled to fulfill the request. The default end date is 7 days from the current date.
- `valid_from` - (Optional) The start date and time of the request, in UTC RFC3339 format (for example, `YYYY-MM-DDTHH:MM:SSZ`). The de-

fault is to start fulfilling the request immediately.

## » Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 10 mins) Used when requesting the spot instance (only valid if `wait_for_fulfillment = true`)
- `delete` - (Defaults to 20 mins) Used when terminating all instances launched via the given spot instance request

## » Attributes Reference

The following attributes are exported:

- `id` - The Spot Instance Request ID.

These attributes are exported, but they are expected to change over time and so should only be used for informational purposes, not for resource dependencies:

- `spot_bid_status` - The current bid status of the Spot Instance Request.
- `spot_request_state` - The current request state of the Spot Instance Request.
- `spot_instance_id` - The Instance ID (if any) that is currently fulfilling the Spot Instance request.
- `public_dns` - The public DNS name assigned to the instance. For EC2-VPC, this is only available if you've enabled DNS hostnames for your VPC
- `public_ip` - The public IP address assigned to the instance, if applicable.
- `private_dns` - The private DNS name assigned to the instance. Can only be used inside the Amazon EC2, and only available if you've enabled DNS hostnames for your VPC
- `private_ip` - The private IP address assigned to the instance

## » `aws__volume__attachment`

Provides an AWS EBS Volume Attachment as a top level resource, to attach and detach volumes from AWS Instances.

**NOTE on EBS block devices:** If you use `ebs_block_device` on an `aws_instance`, Terraform will assume management over the full set of non-root EBS block devices for the instance, and treats additional block devices as drift. For this reason, `ebs_block_device` cannot be mixed with external `aws_ebs_volume` + `aws_ebs_volume_attachment` resources for a given instance.

## » Example Usage

```
resource "aws_volume_attachment" "ebs_att" {
  device_name = "/dev/sdh"
  volume_id   = "${aws_ebs_volume.example.id}"
  instance_id = "${aws_instance.web.id}"
}

resource "aws_instance" "web" {
  ami                = "ami-21f78e11"
  availability_zone  = "us-west-2a"
  instance_type      = "t1.micro"

  tags {
    Name = "HelloWorld"
  }
}

resource "aws_ebs_volume" "example" {
  availability_zone = "us-west-2a"
  size              = 1
}
```

## » Argument Reference

The following arguments are supported:

- `device_name` - (Required) The device name to expose to the instance (for example, `/dev/sdh` or `xvdh`)
- `instance_id` - (Required) ID of the Instance to attach to
- `volume_id` - (Required) ID of the Volume to be attached
- `force_detach` - (Optional, Boolean) Set to `true` if you want to force the volume to detach. Useful if previous attempts failed, but use this option only as a last resort, as this can result in **data loss**. See Detaching an Amazon EBS Volume from an Instance for more information.
- `skip_destroy` - (Optional, Boolean) Set this to `true` if you do not wish to detach the volume from the instance to which it is attached at destroy time, and instead just remove the attachment from Terraform state. This is useful when destroying an instance which has volumes created by some other means attached.

## » Attributes Reference

- `device_name` - The device name exposed to the instance

- `instance_id` - ID of the Instance
- `volume_id` - ID of the Volume

## » `aws_lb`

Provides a Load Balancer resource.

**Note:** `aws_alb` is known as `aws_lb`. The functionality is identical.

## » Example Usage

### » Application Load Balancer

```
resource "aws_lb" "test" {
  name                = "test-lb-tf"
  internal            = false
  load_balancer_type = "application"
  security_groups     = ["${aws_security_group.lb_sg.id}"]
  subnets            = ["${aws_subnet.public.*.id}"]

  enable_deletion_protection = true

  access_logs {
    bucket = "${aws_s3_bucket.lb_logs.bucket}"
    prefix = "test-lb"
    enabled = true
  }

  tags {
    Environment = "production"
  }
}
```

### » Network Load Balancer

```
resource "aws_lb" "test" {
  name                = "test-lb-tf"
  internal            = false
  load_balancer_type = "network"
  subnets           = ["${aws_subnet.public.*.id}"]

  enable_deletion_protection = true
}
```

```

tags {
  Environment = "production"
}
}

```

## » Specifying Elastic IPs

```

resource "aws_lb" "example" {
  name                = "example"
  load_balancer_type = "network"

  subnet_mapping {
    subnet_id      = "${aws_subnet.example1.id}"
    allocation_id = "${aws_eip.example1.id}"
  }

  subnet_mapping {
    subnet_id      = "${aws_subnet.example2.id}"
    allocation_id = "${aws_eip.example2.id}"
  }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The name of the LB. This name must be unique within your AWS account, can have a maximum of 32 characters, must contain only alphanumeric characters or hyphens, and must not begin or end with a hyphen. If not specified, Terraform will autogenerate a name beginning with **tf-lb**.
- **name\_prefix** - (Optional) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **internal** - (Optional) If true, the LB will be internal.
- **load\_balancer\_type** - (Optional) The type of load balancer to create. Possible values are **application** or **network**. The default value is **application**.
- **security\_groups** - (Optional) A list of security group IDs to assign to the LB. Only valid for Load Balancers of type **application**.
- **access\_logs** - (Optional) An Access Logs block. Access Logs documented below. Only valid for Load Balancers of type **application**.
- **subnets** - (Optional) A list of subnet IDs to attach to the LB. Subnets cannot be updated for Load Balancers of type **network**. Changing this value for load balancers of type **network** will force a recreation of the resource.



- **subnet\_mapping** - (Optional) A subnet mapping block as documented below.
- **idle\_timeout** - (Optional) The time in seconds that the connection is allowed to be idle. Only valid for Load Balancers of type **application**. Default: 60.
- **enable\_deletion\_protection** - (Optional) If true, deletion of the load balancer will be disabled via the AWS API. This will prevent Terraform from deleting the load balancer. Defaults to **false**.
- **enable\_cross\_zone\_load\_balancing** - (Optional) If true, cross-zone load balancing of the load balancer will be enabled. This is a **network** load balancer feature. Defaults to **false**.
- **enable\_http2** - (Optional) Indicates whether HTTP/2 is enabled in **application** load balancers. Defaults to **true**.
- **ip\_address\_type** - (Optional) The type of IP addresses used by the subnets for your load balancer. The possible values are **ipv4** and **dualstack**.
- **tags** - (Optional) A mapping of tags to assign to the resource.

**NOTE::** Please note that internal LBs can only use **ipv4** as the **ip\_address\_type**. You can only change to **dualstack** **ip\_address\_type** if the selected subnets are IPv6 enabled.

Access Logs (**access\_logs**) support the following:

- **bucket** - (Required) The S3 bucket name to store the logs in.
- **prefix** - (Optional) The S3 bucket prefix. Logs are stored in the root if not configured.
- **enabled** - (Optional) Boolean to enable / disable **access\_logs**. Defaults to **false**, even when **bucket** is specified.

Subnet Mapping (**subnet\_mapping**) blocks support the following:

- **subnet\_id** - (Required) The id of the subnet of which to attach to the load balancer. You can specify only one subnet per Availability Zone.
- **allocation\_id** - (Optional) The allocation ID of the Elastic IP address.

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- **id** - The ARN of the load balancer (matches **arn**).
- **arn** - The ARN of the load balancer (matches **id**).
- **arn\_suffix** - The ARN suffix for use with CloudWatch Metrics.
- **dns\_name** - The DNS name of the load balancer.
- **canonical\_hosted\_zone\_id** - The canonical hosted zone ID of the load balancer.
- **zone\_id** - The canonical hosted zone ID of the load balancer (to be used in a Route 53 Alias record).

## » Timeouts

`aws_lb` provides the following Timeouts configuration options:

- `create` - (Default 10 minutes) Used for Creating LB
- `update` - (Default 10 minutes) Used for LB modifications
- `delete` - (Default 10 minutes) Used for destroying LB

## » Import

LBs can be imported using their ARN, e.g.

```
$ terraform import aws_lb.bar arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/bar
```

## » aws\_lb\_listener

Provides a Load Balancer Listener resource.

**Note:** `aws_alb_listener` is known as `aws_lb_listener`. The functionality is identical.

## » Example Usage

```
# Create a new load balancer
resource "aws_lb" "front_end" {
  # ...
}

resource "aws_lb_target_group" "front_end" {
  # ...
}

resource "aws_lb_listener" "front_end" {
  load_balancer_arn = "${aws_lb.front_end.arn}"
  port              = "443"
  protocol          = "HTTPS"
  ssl_policy        = "ELBSecurityPolicy-2015-05"
  certificate_arn    = "arn:aws:iam::187416307283:server-certificate/test_cert_rab3wuqwgja25"

  default_action {
    target_group_arn = "${aws_lb_target_group.front_end.arn}"
    type             = "forward"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **load\_balancer\_arn** - (Required, Forces New Resource) The ARN of the load balancer.
- **port** - (Required) The port on which the load balancer is listening.
- **protocol** - (Optional) The protocol for connections from clients to the load balancer. Valid values are `TCP`, `HTTP` and `HTTPS`. Defaults to `HTTP`.
- **ssl\_policy** - (Optional) The name of the SSL Policy for the listener. Required if **protocol** is `HTTPS`.
- **certificate\_arn** - (Optional) The ARN of the default SSL server certificate. Exactly one certificate is required if the protocol is `HTTPS`. For adding additional SSL certificates, see the `aws_lb_listener_certificate` resource.
- **default\_action** - (Required) An Action block. Action blocks are documented below.

Action Blocks (for **default\_action**) support the following:

- **target\_group\_arn** - (Required) The ARN of the Target Group to which to route traffic.
- **type** - (Required) The type of routing action. The only valid value is `forward`.

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- **id** - The ARN of the listener (matches **arn**)
- **arn** - The ARN of the listener (matches **id**)

## » Import

Listeners can be imported using their ARN, e.g.

```
$ terraform import aws_lb_listener.front_end arn:aws:elasticloadbalancing:us-west-2:18741630
```

## » `aws_lb_listener_certificate`

Provides a Load Balancer Listener Certificate resource.

This resource is for additional certificates and does not replace the default certificate on the listener.

**Note:** `aws_alb_listener_certificate` is known as `aws_lb_listener_certificate`. The functionality is identical.

## » Example Usage

```
resource "aws_acm_certificate" "example" {
  # ...
}

resource "aws_lb" "front_end" {
  # ...
}

resource "aws_lb_listener" "front_end" {
  # ...
}

resource "aws_lb_listener_certificate" "example" {
  listener_arn      = "${aws_lb_listener.front_end.arn}"
  certificate_arn = "${aws_acm_certificate.example.arn}"
}
```

## » Argument Reference

The following arguments are supported:

- `listener_arn` - (Required, Forces New Resource) The ARN of the listener to which to attach the certificate.
- `certificate_arn` - (Required, Forces New Resource) The ARN of the certificate to attach to the listener.

## » `aws_lb_listener_rule`

Provides a Load Balancer Listener Rule resource.

**Note:** `aws_alb_listener_rule` is known as `aws_lb_listener_rule`. The functionality is identical.

## » Example Usage

```
# Create a new load balancer
resource "aws_lb" "front_end" {
```

```

    # ...
}

resource "aws_lb_listener" "front_end" {
    # Other parameters
}

resource "aws_lb_listener_rule" "static" {
    listener_arn = "${aws_lb_listener.front_end.arn}"
    priority     = 100

    action {
        type           = "forward"
        target_group_arn = "${aws_lb_target_group.static.arn}"
    }

    condition {
        field = "path-pattern"
        values = ["/static/*"]
    }
}

resource "aws_lb_listener_rule" "host_based_routing" {
    listener_arn = "${aws_lb_listener.front_end.arn}"
    priority     = 99

    action {
        type           = "forward"
        target_group_arn = "${aws_lb_target_group.static.arn}"
    }

    condition {
        field = "host-header"
        values = ["my-service.*.terraform.io"]
    }
}

```

## » Argument Reference

The following arguments are supported:

- **listener\_arn** - (Required, Forces New Resource) The ARN of the listener to which to attach the rule.
- **priority** - (Optional) The priority for the rule between 1 and 50000. Leaving it unset will automatically set the rule with next available priority

after currently existing highest rule. A listener can't have multiple rules with the same priority.

- **action** - (Required) An Action block. Action blocks are documented below.
- **condition** - (Required) A Condition block. Condition blocks are documented below.

Action Blocks (for **action**) support the following:

- **target\_group\_arn** - (Required) The ARN of the Target Group to which to route traffic.
- **type** - (Required) The type of routing action. The only valid value is **forward**.

Condition Blocks (for **condition**) support the following:

- **field** - (Required) The name of the field. Must be one of **path-pattern** for path based routing or **host-header** for host based routing.
- **values** - (Required) The path patterns to match. A maximum of 1 can be defined.

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- **id** - The ARN of the rule (matches **arn**)
- **arn** - The ARN of the rule (matches **id**)

## » Import

Rules can be imported using their ARN, e.g.

```
$ terraform import aws_lb_listener_rule.front_end arn:aws:elasticloadbalancing:us-west-2:187
```

## » aws\_lb\_target\_group

Provides a Target Group resource for use with Load Balancer resources.

**Note:** `aws_alb_target_group` is known as `aws_lb_target_group`. The functionality is identical.

## » Example Usage

```
resource "aws_lb_target_group" "test" {  
  name      = "tf-example-lb-tg"
```

```

    port      = 80
    protocol  = "HTTP"
    vpc_id    = "${aws_vpc.main.id}"
  }

resource "aws_vpc" "main" {
  cidr_block = "10.0.0.0/16"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional, Forces new resource) The name of the target group. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**. Cannot be longer than 6 characters.
- **port** - (Required) The port on which targets receive traffic, unless overridden when registering a specific target.
- **protocol** - (Required) The protocol to use for routing traffic to the targets.
- **vpc\_id** - (Required) The identifier of the VPC in which to create the target group.
- **deregistration\_delay** - (Optional) The amount time for Elastic Load Balancing to wait before changing the state of a deregistering target from draining to unused. The range is 0-3600 seconds. The default value is 300 seconds.
- **stickiness** - (Optional) A Stickiness block. Stickiness blocks are documented below. **stickiness** is only valid if used with Load Balancers of type **Application**
- **health\_check** - (Optional) A Health Check block. Health Check blocks are documented below.
- **target\_type** - (Optional) The type of target that you must specify when registering targets with this target group. The possible values are **instance** (targets are specified by instance ID) or **ip** (targets are specified by IP address). The default is **instance**. Note that you can't specify targets for a target group using both instance IDs and IP addresses. If the target type is **ip**, specify IP addresses from the subnets of the virtual private cloud (VPC) for the target group, the RFC 1918 range (10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16), and the RFC 6598 range (100.64.0.0/10). You can't specify publicly routable IP addresses.
- **tags** - (Optional) A mapping of tags to assign to the resource.

Stickiness Blocks (**stickiness**) support the following:

- **type** - (Required) The type of sticky sessions. The only current possible

value is `lb_cookie`.

- **cookie\_duration** - (Optional) The time period, in seconds, during which requests from a client should be routed to the same target. After this time period expires, the load balancer-generated cookie is considered stale. The range is 1 second to 1 week (604800 seconds). The default value is 1 day (86400 seconds).
- **enabled** - (Optional) Boolean to enable / disable **stickiness**. Default is `true`

**NOTE:** To help facilitate the authoring of modules that support target groups of any protocol, you can define **stickiness** regardless of the protocol chosen. However, for TCP target groups, **enabled** must be **false**.

Health Check Blocks (**health\_check**):

**Note:** The Health Check parameters you can set vary by the **protocol** of the Target Group. Many parameters cannot be set to custom values for **network** load balancers at this time. See [http://docs.aws.amazon.com/elasticloadbalancing/latest/APIReference/API\\_CreateTargetGroup.html](http://docs.aws.amazon.com/elasticloadbalancing/latest/APIReference/API_CreateTargetGroup.html) for a complete reference.

- **interval** - (Optional) The approximate amount of time, in seconds, between health checks of an individual target. Minimum value 5 seconds, Maximum value 300 seconds. Default 30 seconds.
- **path** - (Optional) The destination for the health check request. Default `/`.
- **port** - (Optional) The port to use to connect with the target. Valid values are either ports 1-65536, or **traffic-port**. Defaults to **traffic-port**.
- **protocol** - (Optional) The protocol to use to connect with the target. Defaults to HTTP.
- **timeout** - (Optional) The amount of time, in seconds, during which no response means a failed health check. For Application Load Balancers, the range is 2 to 60 seconds and the default is 5 seconds. For Network Load Balancers, you cannot set a custom value, and the default is 10 seconds for TCP and HTTPS health checks and 6 seconds for HTTP health checks.
- **healthy\_threshold** - (Optional) The number of consecutive health checks successes required before considering an unhealthy target healthy. Defaults to 3.
- **unhealthy\_threshold** - (Optional) The number of consecutive health check failures required before considering the target unhealthy. For Network Load Balancers, this value must be the same as the **healthy\_threshold**. Defaults to 3.
- **matcher** (Optional, only supported on Application Load Balancers): The HTTP codes to use when checking for a successful response from a target. You can specify multiple values (for example, "200,202") or a range of values (for example, "200-299").



## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- `id` - The ARN of the Target Group (matches `arn`)
- `arn` - The ARN of the Target Group (matches `id`)
- `arn_suffix` - The ARN suffix for use with CloudWatch Metrics.
- `name` - The name of the Target Group

## » Import

Target Groups can be imported using their ARN, e.g.

```
$ terraform import aws_lb_target_group.app_front_end arn:aws:elasticloadbalancing:us-west-2:
```

## » `aws_lb_target_group_attachment`

Provides the ability to register instances and containers with a LB target group

**Note:** `aws_alb_target_group_attachment` is known as `aws_lb_target_group_attachment`. The functionality is identical.

## » Example Usage

```
resource "aws_lb_target_group_attachment" "test" {
  target_group_arn = "${aws_lb_target_group.test.arn}"
  target_id        = "${aws_instance.test.id}"
  port             = 80
}

resource "aws_lb_target_group" "test" {
  // Other arguments
}

resource "aws_instance" "test" {
  // Other arguments
}
```

## » Argument Reference

The following arguments are supported:

- **target\_group\_arn** - (Required) The ARN of the target group with which to register targets
- **target\_id** (Required) The ID of the target. This is the Instance ID for an instance, or the container ID for an ECS container. If the target type is ip, specify an IP address.
- **port** - (Optional) The port on which targets receive traffic.
- **availability\_zone** - (Optional) The Availability Zone where the IP address of the target is to be registered.

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- **id** - A unique identifier for the attachment

## » Import

Target Group Attachments cannot be imported.

## » `aws_ecr_lifecycle_policy`

Provides an ECR lifecycle policy.

## » Example Usage

### » Policy on untagged image

```
resource "aws_ecr_repository" "foo" {
  name = "bar"
}

resource "aws_ecr_lifecycle_policy" "foopolicy" {
  repository = "${aws_ecr_repository.foo.name}"

  policy = <<EOF
{
  "rules": [
    {
      "rulePriority": 1,
      "description": "Expire images older than 14 days",
      "selection": {
        "tagStatus": "untagged",
```

```

        "countType": "sinceImagePushed",
        "countUnit": "days",
        "countNumber": 14
    },
    "action": {
        "type": "expire"
    }
}
]
}
EOF
}

```

#### » Policy on tagged image

```

resource "aws_ecr_repository" "foo" {
    name = "bar"
}

resource "aws_ecr_lifecycle_policy" "foopolicy" {
    repository = "${aws_ecr_repository.foo.name}"

    policy = <<EOF
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Keep last 30 images",
            "selection": {
                "tagStatus": "tagged",
                "tagPrefixList": ["v"],
                "countType": "imageCountMoreThan",
                "countNumber": 30
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
EOF
}

```

## » Argument Reference

The following arguments are supported:

- **repository** - (Required) Name of the repository to apply the policy.
- **policy** - (Required) The policy document. This is a JSON formatted string. See more details about Policy Parameters in the official AWS docs.

## » Attributes Reference

The following attributes are exported:

- **repository** - The name of the repository.
- **registry\_id** - The registry ID where the repository was created.

## » Import

ECR Lifecycle Policy can be imported using the name of the repository, e.g.

```
$ terraform import aws_ecr_lifecycle_policy.example tf-example
```

## » `aws__ecr__repository`

Provides an EC2 Container Registry Repository.

**NOTE on ECR Availability:** The EC2 Container Registry is not yet rolled out in all regions - available regions are listed the AWS Docs.

## » Example Usage

```
resource "aws_ecr_repository" "foo" {  
  name = "bar"  
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) Name of the repository.

## » Attributes Reference

The following attributes are exported:

- **arn** - Full ARN of the repository.
- **name** - The name of the repository.
- **registry\_id** - The registry ID where the repository was created.
- **repository\_url** - The URL of the repository (in the form `aws_account_id.dkr.ecr.region.amazonaws.com`).

## » Import

ECR Repositories can be imported using the **name**, e.g.

```
$ terraform import aws_ecr_repository.service test-service
```

## » aws\_ecr\_repository\_policy

Provides an ECR repository policy.

Note that currently only one policy may be applied to a repository.

**NOTE on ECR Availability:** The EC2 Container Registry is not yet rolled out in all regions - available regions are listed the AWS Docs.

## » Example Usage

```
resource "aws_ecr_repository" "foo" {
  name = "bar"
}

resource "aws_ecr_repository_policy" "foopolicy" {
  repository = "${aws_ecr_repository.foo.name}"

  policy = <<EOF
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "new policy",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [
        "ecr:GetDownloadUrlForLayer",
        "ecr:BatchGetImage",

```

```

        "ecr:BatchCheckLayerAvailability",
        "ecr:PutImage",
        "ecr:InitiateLayerUpload",
        "ecr:UploadLayerPart",
        "ecr:CompleteLayerUpload",
        "ecr:DescribeRepositories",
        "ecr:GetRepositoryPolicy",
        "ecr:ListImages",
        "ecr>DeleteRepository",
        "ecr:BatchDeleteImage",
        "ecr:SetRepositoryPolicy",
        "ecr>DeleteRepositoryPolicy"
    ]
}
]
}
EOF
}

```

## » Argument Reference

The following arguments are supported:

- **repository** - (Required) Name of the repository to apply the policy.
- **policy** - (Required) The policy document. This is a JSON formatted string.

## » Attributes Reference

The following attributes are exported:

- **repository** - The name of the repository.
- **registry\_id** - The registry ID where the repository was created.

## » `aws_ecs_cluster`

Provides an ECS cluster.

## » Example Usage

```

resource "aws_ecs_cluster" "foo" {
  name = "white-hart"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the cluster (up to 255 letters, numbers, hyphens, and underscores)

## » Attributes Reference

The following additional attributes are exported:

- **id** - The Amazon Resource Name (ARN) that identifies the cluster
- **arn** - The Amazon Resource Name (ARN) that identifies the cluster

## » Import

ECS clusters can be imported using the **name**, e.g.

```
$ terraform import aws_ecs_cluster.stateless stateless-app
```

## » aws\_\_ecs\_\_service

**Note:** To prevent a race condition during service deletion, make sure to set **depends\_on** to the related **aws\_iam\_role\_policy**; otherwise, the policy may be destroyed too soon and the ECS service will then get stuck in the **DRAINING** state.

Provides an ECS service - effectively a task that is expected to run until an error occurs or a user terminates it (typically a webserver or a database).

See ECS Services section in AWS developer guide.

## » Example Usage

```
resource "aws_ecs_service" "mongo" {
  name           = "mongodb"
  cluster        = "${aws_ecs_cluster.foo.id}"
  task_definition = "${aws_ecs_task_definition.mongo.arn}"
  desired_count  = 3
  iam_role       = "${aws_iam_role.foo.arn}"
  depends_on     = ["aws_iam_role_policy.foo"]

  ordered_placement_strategy {
    type = "binpack"
  }
}
```

```

    field = "cpu"
  }

  load_balancer {
    target_group_arn = "${aws_lb_target_group.foo.arn}"
    container_name   = "mongo"
    container_port   = 8080
  }

  placement_constraints {
    type      = "memberOf"
    expression = "attribute:ecs.availability-zone in [us-west-2a, us-west-2b]"
  }
}

```

## » Ignoring Changes to Desired Count

You can utilize the generic Terraform resource lifecycle configuration block with `ignore_changes` to create an ECS service with an initial count of running instances, then ignore any changes to that count caused externally (e.g. Application Autoscaling).

```

resource "aws_ecs_service" "example" {
  # ... other configurations ...

  # Example: Create service with 2 instances to start
  desired_count = 2

  # Optional: Allow external changes without Terraform plan difference
  lifecycle {
    ignore_changes = ["desired_count"]
  }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the service (up to 255 letters, numbers, hyphens, and underscores)
- **task\_definition** - (Required) The family and revision (`family:revision`) or full ARN of the task definition that you want to run in your service.
- **desired\_count** - (Required) The number of instances of the task definition to place and keep running



- **launch\_type** - (Optional) The launch type on which to run your service. The valid values are **EC2** and **FARGATE**. Defaults to **EC2**.
- **cluster** - (Optional) ARN of an ECS cluster
- **iam\_role** - (Optional) The ARN of IAM role that allows your Amazon ECS container agent to make calls to your load balancer on your behalf. This parameter is only required if you are using a load balancer with your service.
- **deployment\_maximum\_percent** - (Optional) The upper limit (as a percentage of the service's desiredCount) of the number of running tasks that can be running in a service during a deployment.
- **deployment\_minimum\_healthy\_percent** - (Optional) The lower limit (as a percentage of the service's desiredCount) of the number of running tasks that must remain running and healthy in a service during a deployment.
- **placement\_strategy** - (Optional) **Deprecated**, use **ordered\_placement\_strategy** instead.
- **ordered\_placement\_strategy** - (Optional) Service level strategy rules that are taken into consideration during task placement. List from top to bottom in order of precedence. The maximum number of **placement\_strategy** blocks is 5. Defined below.
- **health\_check\_grace\_period\_seconds** - (Optional) Seconds to ignore failing load balancer health checks on newly instantiated tasks to prevent premature shutdown, up to 7200. Only valid for services configured to use load balancers.
- **load\_balancer** - (Optional) A load balancer block. Load balancers documented below.
- **placement\_constraints** - (Optional) rules that are taken into consideration during task placement. Maximum number of **placement\_constraints** is 10. Defined below.
- **network\_configuration** - (Optional) The network configuration for the service. This parameter is required for task definitions that use the awsvpc network mode to receive their own Elastic Network Interface, and it is not supported for other network modes.
- **service\_registries** - (Optional) The service discovery registries for the service. The maximum number of **service\_registries** blocks is 1.

**Note:** As a result of an AWS limitation, a single **load\_balancer** can be attached to the ECS service at most. See related docs.

Load balancers support the following:

- **elb\_name** - (Required for ELB Classic) The name of the ELB (Classic) to associate with the service.
- **target\_group\_arn** - (Required for ALB/NLB) The ARN of the Load Balancer target group to associate with the service.
- **container\_name** - (Required) The name of the container to associate with the load balancer (as it appears in a container definition).
- **container\_port** - (Required) The port on the container to associate with

the load balancer.

## » `ordered_placement_strategy`

`ordered_placement_strategy` supports the following:

- **type** - (Required) The type of placement strategy. Must be one of: `binpack`, `random`, or `spread`
- **field** - (Optional) For the `spread` placement strategy, valid values are `instanceId` (or `host`, which has the same effect), or any platform or custom attribute that is applied to a container instance. For the `binpack` type, valid values are `memory` and `cpu`. For the `random` type, this attribute is not needed. For more information, see Placement Strategy.

**Note:** for `spread`, `host` and `instanceId` will be normalized, by AWS, to be `instanceId`. This means the statefile will show `instanceId` but your config will differ if you use `host`.

## » `placement_constraints`

`placement_constraints` support the following:

- **type** - (Required) The type of constraint. The only valid values at this time are `memberOf` and `distinctInstance`.
- **expression** - (Optional) Cluster Query Language expression to apply to the constraint. Does not need to be specified for the `distinctInstance` type. For more information, see Cluster Query Language in the Amazon EC2 Container Service Developer Guide.

## » `network_configuration`

`network_configuration` support the following:

- **subnets** - (Required) The subnets associated with the task or service.
- **security\_groups** - (Optional) The security groups associated with the task or service. If you do not specify a security group, the default security group for the VPC is used.
- **assign\_public\_ip** - (Optional) Assign a public IP address to the ENI (Fargate launch type only). Valid values are `true` or `false`. Default `false`.

For more information, see Task Networking

## » `service__registries`

`service_registries` support the following:

- **registry\_arn** - (Required) The ARN of the Service Registry. The currently supported service registry is Amazon Route 53 Auto Naming Service(`aws_service_discovery_service`). For more information, see Service
- **port** - (Optional) The port value used if your Service Discovery service specified an SRV record.

## » Attributes Reference

The following attributes are exported:

- **id** - The Amazon Resource Name (ARN) that identifies the service
- **name** - The name of the service
- **cluster** - The Amazon Resource Name (ARN) of cluster which the service runs on
- **iam\_role** - The ARN of IAM role used for ELB
- **desired\_count** - The number of instances of the task definition

## » Import

ECS services can be imported using the **name** together with ecs cluster **name**, e.g.

```
$ terraform import aws_ecs_service.imported cluster-name/service-name
```

## » `aws__ecs__task__definition`

Provides an ECS task definition to be used in `aws_ecs_service`.

## » Example Usage

```
resource "aws_ecs_task_definition" "service" {
  family           = "service"
  container_definitions = "${file("task-definitions/service.json")}"

  volume {
    name      = "service-storage"
    host_path = "/ecs/service-storage"
  }
}
```

```

placement_constraints {
  type      = "memberOf"
  expression = "attribute:ecs.availability-zone in [us-west-2a, us-west-2b]"
}
}

```

The referenced `task-definitions/service.json` file contains a valid JSON document, which is shown below, and its content is going to be passed directly into the `container_definitions` attribute as a string. Please note that this example contains only a small subset of the available parameters.

```

[
  {
    "name": "first",
    "image": "service-first",
    "cpu": 10,
    "memory": 512,
    "essential": true,
    "portMappings": [
      {
        "containerPort": 80,
        "hostPort": 80
      }
    ]
  },
  {
    "name": "second",
    "image": "service-second",
    "cpu": 10,
    "memory": 256,
    "essential": true,
    "portMappings": [
      {
        "containerPort": 443,
        "hostPort": 443
      }
    ]
  }
]

```

## » Argument Reference

### » Top-Level Arguments

- `family` - (Required) A unique name for your task definition.

- **container\_definitions** - (Required) A list of valid container definitions provided as a single valid JSON document. Please note that you should only provide values that are part of the container definition document. For a detailed description of what parameters are available, see the Task Definition Parameters section from the official Developer Guide.

**NOTE:** Proper escaping is required for JSON field values containing quotes (") such as **environment** values. If directly setting the JSON, they should be escaped as `\` in the JSON, e.g. `"value": "I \"love\" escaped quotes"`. If using a Terraform variable value, they should be escaped as `\\` in the variable, e.g. `value = "I \\\"love\\\" escaped quotes"` in the variable and `"value": "${var.myvariable}"` in the JSON.

- **task\_role\_arn** - (Optional) The ARN of IAM role that allows your Amazon ECS container task to make calls to other AWS services.
- **execution\_role\_arn** - (Optional) The Amazon Resource Name (ARN) of the task execution role that the Amazon ECS container agent and the Docker daemon can assume.
- **network\_mode** - (Optional) The Docker networking mode to use for the containers in the task. The valid values are **none**, **bridge**, **awsvpc**, and **host**.
- **volume** - (Optional) A set of volume blocks that containers in your task may use.
- **placement\_constraints** - (Optional) A set of placement constraints rules that are taken into consideration during task placement. Maximum number of **placement\_constraints** is 10.
- **cpu** - (Optional) The number of cpu units used by the task. If the **requires\_compatibilities** is **FARGATE** this field is required.
- **memory** - (Optional) The amount (in MiB) of memory used by the task. If the **requires\_compatibilities** is **FARGATE** this field is required.
- **requires\_compatibilities** - (Optional) A set of launch types required by the task. The valid values are **EC2** and **FARGATE**.

## » Volume Block Arguments

- **name** - (Required) The name of the volume. This name is referenced in the **sourceVolume** parameter of container definition in the **mountPoints** section.
- **host\_path** - (Optional) The path on the host container instance that is presented to the container. If not set, ECS will create a nonpersistent data volume that starts empty and is deleted after the task has finished.

## » Placement Constraints Arguments

- **type** - (Required) The type of constraint. Use **memberOf** to restrict selection to a group of valid candidates. Note that **distinctInstance** is not

supported in task definitions.

- **expression** - (Optional) Cluster Query Language expression to apply to the constraint. For more information, see Cluster Query Language in the Amazon EC2 Container Service Developer Guide.

## » Attributes Reference

The following attributes are exported:

- **arn** - Full ARN of the Task Definition (including both **family** and **revision**).
- **family** - The family of the Task Definition.
- **revision** - The revision of the task in a particular family.

## » `aws_efs_file_system`

Provides an Elastic File System (EFS) resource.

## » Example Usage

```
resource "aws_efs_file_system" "foo" {
  creation_token = "my-product"

  tags {
    Name = "MyProduct"
  }
}
```

## » Argument Reference

**NOTE:** The `reference_name` attribute has been deprecated and might be removed in future releases, please use `creation_token` instead.

The following arguments are supported:

- **creation\_token** - (Optional) A unique name (a maximum of 64 characters are allowed) used as reference when creating the Elastic File System to ensure idempotent file system creation. By default generated by Terraform. See Elastic File System user guide for more information.
- **reference\_name** - **DEPRECATED** (Optional) A reference name used when creating the **Creation Token** which Amazon EFS uses to ensure idempotent file system creation. By default generated by Terraform.

- **performance\_mode** - (Optional) The file system performance mode. Can be either "generalPurpose" or "maxIO" (Default: "generalPurpose").
- **tags** - (Optional) A mapping of tags to assign to the file system.
- **encrypted** - (Optional) If true, the disk will be encrypted.
- **kms\_key\_id** - (Optional) The ARN for the KMS encryption key. When specifying kms\_key\_id, encrypted needs to be set to true.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID that identifies the file system (e.g. fs-ccfc0d65).
- **kms\_key\_id** - The ARN for the KMS encryption key.
- **dns\_name** - The DNS name for the filesystem per documented convention.

## » Import

The EFS file systems can be imported using the **id**, e.g.

```
$ terraform import aws_efs_file_system.foo fs-6fa144c6
```

## » aws\_efs\_mount\_target

Provides an Elastic File System (EFS) mount target.

## » Example Usage

```
resource "aws_efs_mount_target" "alpha" {
  file_system_id = "${aws_efs_file_system.foo.id}"
  subnet_id      = "${aws_subnet.alpha.id}"
}

resource "aws_vpc" "foo" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_subnet" "alpha" {
  vpc_id            = "${aws_vpc.foo.id}"
  availability_zone = "us-west-2a"
  cidr_block        = "10.0.1.0/24"
}
```

## » Argument Reference

The following arguments are supported:

- **file\_system\_id** - (Required) The ID of the file system for which the mount target is intended.
- **subnet\_id** - (Required) The ID of the subnet to add the mount target in.
- **ip\_address** - (Optional) The address (within the address range of the specified subnet) at which the file system may be mounted via the mount target.
- **security\_groups** - (Optional) A list of up to 5 VPC security group IDs (that must be for the same VPC as subnet specified) in effect for the mount target.

## » Attributes Reference

**Note:** The **dns\_name** attribute is only useful if the mount target is in a VPC that has support for DNS hostnames enabled. See [Using DNS with Your VPC and VPC resource in Terraform](#) for more information.

In addition to all arguments above, the following attributes are exported:

- **id** - The ID of the mount target.
- **dns\_name** - The DNS name for the given subnet/AZ per documented convention.
- **network\_interface\_id** - The ID of the network interface that Amazon EFS created when it created the mount target.

## » Import

The EFS mount targets can be imported using the **id**, e.g.

```
$ terraform import aws_efs_mount_target.alpha fsmt-52a643fb
```

## » aws\_\_elasticsearch\_\_cluster

Provides an ElastiCache Cluster resource, which manages a Memcached cluster or Redis instance. For working with Redis (Cluster Mode Enabled) replication groups, see the **aws\_elasticsearch\_replication\_group** resource.

**Note:** When you change an attribute, such as **node\_type**, by default it is applied in the next maintenance window. Because of this, Terraform may report a difference in its planning phase because the actual modification has not yet taken place. You can use the **apply\_immediately** flag to instruct the service to apply the change immediately. Using **apply\_immediately** can result in a brief



downtime as the server reboots. See the AWS Docs on Modifying an ElastiCache Cache Cluster for more information.

## » Example Usage

### » Memcached Cluster

```
resource "aws_elasticache_cluster" "example" {
  cluster_id      = "cluster-example"
  engine          = "memcached"
  node_type       = "cache.m3.medium"
  num_cache_nodes = 2
  parameter_group_name = "default.memcached1.4"
  port           = 11211
}
```

### » Redis Instance

```
resource "aws_elasticache_cluster" "example" {
  cluster_id      = "cluster-example"
  engine          = "redis"
  node_type       = "cache.m3.medium"
  num_cache_nodes = 1
  parameter_group_name = "default.redis3.2"
  port           = 6379
}
```

### » Redis Cluster Mode Disabled Read Replica Instance

These inherit their settings from the replication group.

```
resource "aws_elasticache_cluster" "replica" {
  cluster_id            = "cluster-example"
  replication_group_id = "${aws_elasticache_replication_group.example.id}"
}
```

## » Argument Reference

The following arguments are supported:

- `cluster_id` – (Required) Group identifier. ElastiCache converts this name to lowercase

- **replication\_group\_id** - (Optional) The ID of the replication group to which this cluster should belong. If this parameter is specified, the cluster is added to the specified replication group as a read replica; otherwise, the cluster is a standalone primary that is not part of any replication group.
- **engine** - (Required unless **replication\_group\_id** is provided) Name of the cache engine to be used for this cache cluster. Valid values for this parameter are **memcached** or **redis**
- **engine\_version** - (Optional) Version number of the cache engine to be used. See Selecting a Cache Engine and Version in the AWS Documentation center for supported versions
- **maintenance\_window** - (Optional) Specifies the weekly time range for when maintenance on the cache cluster is performed. The format is **ddd:hh24:mi-ddd:hh24:mi** (24H Clock UTC). The minimum maintenance window is a 60 minute period. Example: **sun:05:00-sun:09:00**
- **node\_type** - (Required unless **replication\_group\_id** is provided) The compute and memory capacity of the nodes. See Available Cache Node Types for supported node types
- **num\_cache\_nodes** - (Required unless **replication\_group\_id** is provided) The initial number of cache nodes that the cache cluster will have. For Redis, this value must be 1. For Memcache, this value must be between 1 and 20. If this number is reduced on subsequent runs, the highest numbered nodes will be removed.
- **parameter\_group\_name** - (Required unless **replication\_group\_id** is provided) Name of the parameter group to associate with this cache cluster
- **port** - (Optional) The port number on which each of the cache nodes will accept connections. For Memcache the default is 11211, and for Redis the default port is 6379. Cannot be provided with **replication\_group\_id**.
- **subnet\_group\_name** - (Optional, VPC only) Name of the subnet group to be used for the cache cluster.
- **security\_group\_names** - (Optional, EC2 Classic only) List of security group names to associate with this cache cluster
- **security\_group\_ids** - (Optional, VPC only) One or more VPC security groups associated with the cache cluster
- **apply\_immediately** - (Optional) Specifies whether any database modifications are applied immediately, or during the next maintenance window. Default is **false**. See Amazon ElastiCache Documentation for more information. (Available since v0.6.0)
- **snapshot\_arns** - (Optional) A single-element string list containing an Amazon Resource Name (ARN) of a Redis RDB snapshot file stored in

Amazon S3. Example: `arn:aws:s3:::my_bucket/snapshot1.rdb`

- **snapshot\_name** - (Optional) The name of a snapshot from which to restore data into the new node group. Changing the **snapshot\_name** forces a new resource.
- **snapshot\_window** - (Optional, Redis only) The daily time range (in UTC) during which ElastiCache will begin taking a daily snapshot of your cache cluster. Example: 05:00-09:00
- **snapshot\_retention\_limit** - (Optional, Redis only) The number of days for which ElastiCache will retain automatic cache cluster snapshots before deleting them. For example, if you set `SnapshotRetentionLimit` to 5, then a snapshot that was taken today will be retained for 5 days before being deleted. If the value of `SnapshotRetentionLimit` is set to zero (0), backups are turned off. Please note that setting a **snapshot\_retention\_limit** is not supported on `cache.t1.micro` or `cache.t2.*` cache nodes
- **notification\_topic\_arn** - (Optional) An Amazon Resource Name (ARN) of an SNS topic to send ElastiCache notifications to. Example: `arn:aws:sns:us-east-1:012345678999:my_sns_topic`
- **az\_mode** - (Optional, Memcached only) Specifies whether the nodes in this Memcached node group are created in a single Availability Zone or created across multiple Availability Zones in the cluster's region. Valid values for this parameter are `single-az` or `cross-az`, default is `single-az`. If you want to choose `cross-az`, **num\_cache\_nodes** must be greater than 1
- **availability\_zone** - (Optional) The Availability Zone for the cache cluster. If you want to create cache nodes in multi-az, use **availability\_zones**
- **availability\_zones** - (Optional, Memcached only) List of Availability Zones in which the cache nodes will be created. If you want to create cache nodes in single-az, use **availability\_zone**
- **tags** - (Optional) A mapping of tags to assign to the resource

## » Attributes Reference

The following additional attributes are exported:

- **cache\_nodes** - List of node objects including `id`, `address`, `port` and `availability_zone`. Referenceable e.g. as `${aws_elasticache_cluster.bar.cache_nodes.0.address}`
- **configuration\_endpoint** - (Memcached only) The configuration endpoint to allow host discovery.
- **cluster\_address** - (Memcached only) The DNS name of the cache cluster without the port appended.

## » Import

ElastiCache Clusters can be imported using the `cluster_id`, e.g.

```
$ terraform import aws_elasticache_cluster.my_cluster my_cluster
```

## » `aws_elasticache_parameter_group`

Provides an ElastiCache parameter group resource.

## » Example Usage

```
resource "aws_elasticache_parameter_group" "default" {
  name      = "cache-params"
  family    = "redis2.8"

  parameter {
    name = "activeresharding"
    value = "yes"
  }

  parameter {
    name = "min-slaves-to-write"
    value = "2"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the ElastiCache parameter group.
- **family** - (Required) The family of the ElastiCache parameter group.
- **description** - (Optional) The description of the ElastiCache parameter group. Defaults to "Managed by Terraform".
- **parameter** - (Optional) A list of ElastiCache parameters to apply.

Parameter blocks support the following:

- **name** - (Required) The name of the ElastiCache parameter.
- **value** - (Required) The value of the ElastiCache parameter.

## » Attributes Reference

The following attributes are exported:

- `id` - The ElastiCache parameter group name.

## » Import

ElastiCache Parameter Groups can be imported using the `name`, e.g.

```
$ terraform import aws_elasticache_parameter_group.default redis-params
```

## » `aws_elasticache_replication_group`

Provides an ElastiCache Replication Group resource. For working with Memcached or single primary Redis instances (Cluster Mode Disabled), see the `aws_elasticache_cluster` resource.

## » Example Usage

### » Redis Cluster Mode Disabled

To create a single shard primary with single read replica:

```
resource "aws_elasticache_replication_group" "example" {
  automatic_failover_enabled = true
  availability_zones         = ["us-west-2a", "us-west-2b"]
  replication_group_id       = "tf-rep-group-1"
  replication_group_description = "test description"
  node_type                  = "cache.m3.medium"
  number_cache_clusters      = 2
  parameter_group_name       = "default.redis3.2"
  port                       = 6379
}
```

Additional read replicas can be added or removed with the `aws_elasticache_cluster` resource and its `replication_group_id` attribute. In this situation, you will need to utilize the lifecycle configuration block with `ignore_changes` to prevent replication group recreation.

```
resource "aws_elasticache_replication_group" "example" {
  automatic_failover_enabled = true
  availability_zones         = ["us-west-2a", "us-west-2b"]
  replication_group_id       = "tf-rep-group-1"
  replication_group_description = "test description"
```

```

node_type           = "cache.m3.medium"
number_cache_clusters = 2
parameter_group_name = "default.redis3.2"
port                = 6379

lifecycle {
  ignore_changes = ["number_cache_clusters"]
}
}

resource "aws_elasticache_cluster" "replica" {
  count = 1

  cluster_id          = "tf-rep-group-1-${count.index}"
  replication_group_id = "${aws_elasticache_replication_group.example.id}"
}

```

## » Redis Cluster Mode Enabled

To create two shards with a primary and a single read replica each:

```

resource "aws_elasticache_replication_group" "baz" {
  replication_group_id          = "tf-redis-cluster"
  replication_group_description = "test description"
  node_type                     = "cache.m1.small"
  port                          = 6379
  parameter_group_name          = "default.redis3.2.cluster.on"
  automatic_failover_enabled    = true
  cluster_mode {
    replicas_per_node_group = 1
    num_node_groups         = 2
  }
}

```

**Note:** We currently do not support passing a `primary_cluster_id` in order to create the Replication Group.

**Note:** Automatic Failover is unavailable for Redis versions earlier than 2.8.6, and unavailable on T1 and T2 node types. See the Amazon Replication with Redis guide for full details on using Replication Groups.

## » Argument Reference

The following arguments are supported:

- **replication\_group\_id** - (Required) The replication group identifier. This parameter is stored as a lowercase string.
- **replication\_group\_description** - (Required) A user-created description for the replication group.
- **number\_cache\_clusters** - (Required) The number of cache clusters this replication group will have. If Multi-AZ is enabled, the value of this parameter must be at least 2. Changing this number will force a new resource
- **node\_type** - (Required) The compute and memory capacity of the nodes in the node group.
- **automatic\_failover\_enabled** - (Optional) Specifies whether a read-only replica will be automatically promoted to read/write primary if the existing primary fails. If true, Multi-AZ is enabled for this replication group. If false, Multi-AZ is disabled for this replication group. Must be enabled for Redis (cluster mode enabled) replication groups. Defaults to **false**.
- **auto\_minor\_version\_upgrade** - (Optional) Specifies whether a minor engine upgrades will be applied automatically to the underlying Cache Cluster instances during the maintenance window. Defaults to **true**.
- **availability\_zones** - (Optional) A list of EC2 availability zones in which the replication group's cache clusters will be created. The order of the availability zones in the list is not important.
- **engine** - (Optional) The name of the cache engine to be used for the clusters in this replication group. e.g. **redis**
- **at\_rest\_encryption\_enabled** - (Optional) Whether to enable encryption at rest.
- **transit\_encryption\_enabled** - (Optional) Whether to enable encryption in transit.
- **auth\_token** - (Optional) The password used to access a password protected server. Can be specified only if **transit\_encryption\_enabled = true**.
- **engine\_version** - (Optional) The version number of the cache engine to be used for the cache clusters in this replication group.
- **parameter\_group\_name** - (Optional) The name of the parameter group to associate with this replication group. If this argument is omitted, the default cache parameter group for the specified engine is used.
- **port** - (Optional) The port number on which each of the cache nodes will accept connections. For Memcache the default is 11211, and for Redis the default port is 6379.
- **subnet\_group\_name** - (Optional) The name of the cache subnet group to be used for the replication group.
- **security\_group\_names** - (Optional) A list of cache security group names to associate with this replication group.
- **security\_group\_ids** - (Optional) One or more Amazon VPC security groups associated with this replication group. Use this parameter only when you are creating a replication group in an Amazon Virtual Private Cloud

- **snapshot\_arns** - (Optional) A single-element string list containing an Amazon Resource Name (ARN) of a Redis RDB snapshot file stored in Amazon S3. Example: `arn:aws:s3:::my_bucket/snapshot1.rdb`
- **snapshot\_name** - (Optional) The name of a snapshot from which to restore data into the new node group. Changing the **snapshot\_name** forces a new resource.
- **maintenance\_window** - (Optional) Specifies the weekly time range for when maintenance on the cache cluster is performed. The format is `ddd:hh24:mi-ddd:hh24:mi` (24H Clock UTC). The minimum maintenance window is a 60 minute period. Example: `sun:05:00-sun:09:00`
- **notification\_topic\_arn** - (Optional) An Amazon Resource Name (ARN) of an SNS topic to send ElastiCache notifications to. Example: `arn:aws:sns:us-east-1:012345678999:my_sns_topic`
- **snapshot\_window** - (Optional, Redis only) The daily time range (in UTC) during which ElastiCache will begin taking a daily snapshot of your cache cluster. The minimum snapshot window is a 60 minute period. Example: `05:00-09:00`
- **snapshot\_retention\_limit** - (Optional, Redis only) The number of days for which ElastiCache will retain automatic cache cluster snapshots before deleting them. For example, if you set `SnapshotRetentionLimit` to 5, then a snapshot that was taken today will be retained for 5 days before being deleted. If the value of `SnapshotRetentionLimit` is set to zero (0), backups are turned off. Please note that setting a **snapshot\_retention\_limit** is not supported on `cache.t1.micro` or `cache.t2.*` cache nodes
- **apply\_immediately** - (Optional) Specifies whether any modifications are applied immediately, or during the next maintenance window. Default is `false`.
- **tags** - (Optional) A mapping of tags to assign to the resource
- **cluster\_mode** - (Optional) Create a native redis cluster. **automatic\_failover\_enabled** must be set to true. Cluster Mode documented below. Only 1 **cluster\_mode** block is allowed.

Cluster Mode (**cluster\_mode**) supports the following:

- **replicas\_per\_node\_group** - (Required) Specify the number of replica nodes in each node group. Valid values are 0 to 5. Changing this number will force a new resource.
- **num\_node\_groups** - (Required) Specify the number of node groups (shards) for this Redis replication group. Changing this number will trigger an online resizing operation before other settings modifications.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the ElastiCache Replication Group.



- `configuration_endpoint_address` - The address of the replication group configuration endpoint when cluster mode is enabled.
- `primary_endpoint_address` - (Redis only) The address of the endpoint for the primary node in the replication group, if the cluster mode is disabled.

## » Timeouts

`aws_elasticache_replication_group` provides the following Timeouts configuration options:

- `create` - (Default 60m) How long to wait for a replication group to be created.
- `delete` - (Default 40m) How long to wait for a replication group to be deleted.
- `update` - (Default 40m) How long to wait for replication group settings to be updated. This is also separately used for online resize operation completion, if necessary.

## » Import

ElastiCache Replication Groups can be imported using the `replication_group_id`, e.g.

```
$ terraform import aws_elasticache_replication_group.my_replication_group replication-group-
```

## » `aws__elasticache__security__group`

Provides an ElastiCache Security Group to control access to one or more cache clusters.

**NOTE:** ElastiCache Security Groups are for use only when working with an ElastiCache cluster **outside** of a VPC. If you are using a VPC, see the ElastiCache Subnet Group resource.

## » Example Usage

```
resource "aws_security_group" "bar" {
  name = "security-group"
}

resource "aws_elasticache_security_group" "bar" {
  name = "elasticache-security-group"
```

```

    security_group_names = ["${aws_security_group.bar.name}"]
}

```

## » Argument Reference

The following arguments are supported:

- **name** – (Required) Name for the cache security group. This value is stored as a lowercase string.
- **description** – (Optional) description for the cache security group. Defaults to "Managed by Terraform".
- **security\_group\_names** – (Required) List of EC2 security group names to be authorized for ingress to the cache security group

## » Attributes Reference

The following attributes are exported:

- **description**
- **name**
- **security\_group\_names**

## » Import

ElastiCache Security Groups can be imported by name, e.g.

```
$ terraform import aws_elasticache_security_group.my_ec_security_group ec-security-group-1
```

## » aws\_\_elasticache\_\_subnet\_\_group

Provides an ElastiCache Subnet Group resource.

**NOTE:** ElastiCache Subnet Groups are only for use when working with an ElastiCache cluster **inside** of a VPC. If you are on EC2 Classic, see the ElastiCache Security Group resource.

## » Example Usage

```

resource "aws_vpc" "foo" {
  cidr_block = "10.0.0.0/16"

  tags {
    Name = "tf-test"
  }
}

```

```

    }
  }

resource "aws_subnet" "foo" {
  vpc_id            = "${aws_vpc.foo.id}"
  cidr_block        = "10.0.0.0/24"
  availability_zone = "us-west-2a"

  tags {
    Name = "tf-test"
  }
}

resource "aws_elasticache_subnet_group" "bar" {
  name          = "tf-test-cache-subnet"
  subnet_ids    = ["${aws_subnet.foo.id}"]
}

```

## » Argument Reference

The following arguments are supported:

- **name** – (Required) Name for the cache subnet group. Elasticache converts this name to lowercase.
- **description** – (Optional) Description for the cache subnet group. Defaults to "Managed by Terraform".
- **subnet\_ids** – (Required) List of VPC Subnet IDs for the cache subnet group

## » Attributes Reference

The following attributes are exported:

- **description**
- **name**
- **subnet\_ids**

## » Import

ElastiCache Subnet Groups can be imported using the **name**, e.g.

```
$ terraform import aws_elasticache_subnet_group.bar tf-test-cache-subnet
```

## » `aws_elastic_beanstalk_application`

Provides an Elastic Beanstalk Application Resource. Elastic Beanstalk allows you to deploy and manage applications in the AWS cloud without worrying about the infrastructure that runs those applications.

This resource creates an application that has one configuration template named `default`, and no application versions

### » Example Usage

```
resource "aws_elastic_beanstalk_application" "tftest" {
  name          = "tf-test-name"
  description    = "tf-test-desc"

  appversion_lifecycle {
    service_role      = "${aws_iam_role.beanstalk_service.arn}"
    max_count         = 128
    delete_source_from_s3 = true
  }
}
```

### » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the application, must be unique within your account
- **description** - (Optional) Short description of the application

Application version lifecycle (`appversion_lifecycle`) supports the following settings. Only one of either `max_count` or `max_age_in_days` can be provided:

- **service\_role** - (Required) The ARN of an IAM service role under which the application version is deleted. Elastic Beanstalk must have permission to assume this role.
- **max\_count** - (Optional) The maximum number of application versions to retain.
- **max\_age\_in\_days** - (Optional) The number of days to retain an application version.
- **delete\_source\_from\_s3** - (Optional) Set to `true` to delete a version's source bundle from S3 when the application version is deleted.

## » Attributes Reference

The following attributes are exported:

- name
- description

## » Import

Elastic Beanstalk Applications can be imported using the **name**, e.g.

```
$ terraform import aws_elastic_beanstalk_application.tf_test tf-test-name
```

## » aws\_elastic\_beanstalk\_application\_version

Provides an Elastic Beanstalk Application Version Resource. Elastic Beanstalk allows you to deploy and manage applications in the AWS cloud without worrying about the infrastructure that runs those applications.

This resource creates a Beanstalk Application Version that can be deployed to a Beanstalk Environment.

**NOTE on Application Version Resource:** When using the Application Version resource with multiple Elastic Beanstalk Environments it is possible that an error may be returned when attempting to delete an Application Version while it is still in use by a different environment. To work around this you can:

1. Create each environment in a separate AWS account
2. Create your `aws_elastic_beanstalk_application_version` resources with a unique names in your Elastic Beanstalk Application. For example `<revision>-<environment>`.

## » Example Usage

```
resource "aws_s3_bucket" "default" {
  bucket = "tftest.applicationversion.bucket"
}

resource "aws_s3_bucket_object" "default" {
  bucket = "${aws_s3_bucket.default.id}"
  key    = "beanstalk/go-v1.zip"
  source = "go-v1.zip"
}

resource "aws_elastic_beanstalk_application" "default" {
```

```

    name          = "tf-test-name"
    description    = "tf-test-desc"
  }

resource "aws_elastic_beanstalk_application_version" "default" {
  name          = "tf-test-version-label"
  application    = "tf-test-name"
  description    = "application version created by terraform"
  bucket        = "${aws_s3_bucket.default.id}"
  key           = "${aws_s3_bucket_object.default.id}"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A unique name for the this Application Version.
- **application** - (Required) Name of the Beanstalk Application the version is associated with.
- **description** - (Optional) Short description of the Application Version.
- **bucket** - (Required) S3 bucket that contains the Application Version source bundle.
- **key** - (Required) S3 object that is the Application Version source bundle.
- **force\_delete** - (Optional) On delete, force an Application Version to be deleted when it may be in use by multiple Elastic Beanstalk Environments.

## » Attributes Reference

The following attributes are exported:

- **name** - The Application Version name.

## » aws\_elastic\_beanstalk\_configuration\_template

Provides an Elastic Beanstalk Configuration Template, which are associated with a specific application and are used to deploy different versions of the application with the same configuration settings.

## » Example Usage

```

resource "aws_elastic_beanstalk_application" "tftest" {
  name          = "tf-test-name"
}

```

```

    description = "tf-test-desc"
}

resource "aws_elastic_beanstalk_configuration_template" "tf_template" {
  name           = "tf-test-template-config"
  application     = "${aws_elastic_beanstalk_application.tf_test.name}"
  solution_stack_name = "64bit Amazon Linux 2015.09 v2.0.8 running Go 1.4"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A unique name for this Template.
- **application** - (Required) name of the application to associate with this configuration template
- **description** - (Optional) Short description of the Template
- **environment\_id** - (Optional) The ID of the environment used with this configuration template
- **setting** - (Optional) Option settings to configure the new Environment. These override specific values that are set as defaults. The format is detailed below in Option Settings
- **solution\_stack\_name** - (Optional) A solution stack to base your Template off of. Example stacks can be found in the Amazon API documentation

## » Option Settings

The **setting** field supports the following format:

- **namespace** - unique namespace identifying the option's associated AWS resource
- **name** - name of the configuration option
- **value** - value for the configuration option
- **resource** - (Optional) resource name for scheduled action

## » Attributes Reference

The following attributes are exported:

- **name**
- **application**
- **description**
- **environment\_id**

- `option_settings`
- `solution_stack_name`

## » `aws_elastic_beanstalk_environment`

Provides an Elastic Beanstalk Environment Resource. Elastic Beanstalk allows you to deploy and manage applications in the AWS cloud without worrying about the infrastructure that runs those applications.

Environments are often things such as `development`, `integration`, or `production`.

### » Example Usage

```
resource "aws_elastic_beanstalk_application" "tftest" {
  name          = "tf-test-name"
  description   = "tf-test-desc"
}

resource "aws_elastic_beanstalk_environment" "tfenvtest" {
  name          = "tf-test-name"
  application    = "${aws_elastic_beanstalk_application.tftest.name}"
  solution_stack_name = "64bit Amazon Linux 2015.03 v2.0.3 running Go 1.4"
}
```

### » Argument Reference

The following arguments are supported:

- **name** - (Required) A unique name for this Environment. This name is used in the application URL
- **application** - (Required) Name of the application that contains the version to be deployed
- **cname\_prefix** - (Optional) Prefix to use for the fully qualified DNS name of the Environment.
- **description** - (Optional) Short description of the Environment
- **tier** - (Optional) Elastic Beanstalk Environment tier. Valid values are `Worker` or `WebServer`. If tier is left blank `WebServer` will be used.
- **setting** - (Optional) Option settings to configure the new Environment. These override specific values that are set as defaults. The format is detailed below in Option Settings



- **solution\_stack\_name** – (Optional) A solution stack to base your environment off of. Example stacks can be found in the Amazon API documentation
- **template\_name** – (Optional) The name of the Elastic Beanstalk Configuration template to use in deployment
- **wait\_for\_ready\_timeout** - (Default: 20m) The maximum duration that Terraform should wait for an Elastic Beanstalk Environment to be in a ready state before timing out.
- **poll\_interval** – The time between polling the AWS API to check if changes have been applied. Use this to adjust the rate of API calls for any **create** or **update** action. Minimum 10s, maximum 180s. Omit this to use the default behavior, which is an exponential backoff
- **version\_label** - (Optional) The name of the Elastic Beanstalk Application Version to use in deployment.
- **tags** – (Optional) A set of tags to apply to the Environment.

## » Option Settings

Some options can be stack-specific, check AWS Docs for supported options and examples.

The **setting** and **all\_settings** mappings support the following format:

- **namespace** - unique namespace identifying the option's associated AWS resource
- **name** - name of the configuration option
- **value** - value for the configuration option
- **resource** - (Optional) resource name for scheduled action

## » Example With Options

```
resource "aws_elastic_beanstalk_application" "tfctest" {
  name          = "tf-test-name"
  description = "tf-test-desc"
}

resource "aws_elastic_beanstalk_environment" "tfenvtest" {
  name          = "tf-test-name"
  application    = "${aws_elastic_beanstalk_application.tfctest.name}"
  solution_stack_name = "64bit Amazon Linux 2015.03 v2.0.3 running Go 1.4"

  setting {
    namespace = "aws:ec2:vpc"
    name      = "VPCId"
    value     = "vpc-xxxxxxx"
  }
}
```

```

    }

    setting {
      namespace = "aws:ec2:vpc"
      name      = "Subnets"
      value     = "subnet-xxxxxxx"
    }
  }
}

```

## » Attributes Reference

The following attributes are exported:

- **id** - ID of the Elastic Beanstalk Environment.
- **name** - Name of the Elastic Beanstalk Environment.
- **description** - Description of the Elastic Beanstalk Environment.
- **tier** - The environment tier specified.
- **application** - The Elastic Beanstalk Application specified for this environment.
- **setting** - Settings specifically set for this Environment.
- **all\_settings** - List of all option settings configured in the Environment. These are a combination of default settings and their overrides from **setting** in the configuration.
- **cname** - Fully qualified DNS name for the Environment.
- **autoscaling\_groups** - The autoscaling groups used by this environment.
- **instances** - Instances used by this environment.
- **launch\_configurations** - Launch configurations in use by this environment.
- **load\_balancers** - Elastic load balancers in use by this environment.
- **queues** - SQS queues in use by this environment.
- **triggers** - Autoscaling triggers in use by this environment.

## » Import

Elastic Beanstalk Environments can be imported using the **id**, e.g.

```
$ terraform import aws_elastic_beanstalk_environment.prodenv e-rpqsewtp2j
```

## » `aws__emr__cluster`

Provides an Elastic MapReduce Cluster, a web service that makes it easy to process large amounts of data efficiently. See Amazon Elastic MapReduce Documentation for more information.

## » Example Usage

```
resource "aws_emr_cluster" "emr-test-cluster" {
  name           = "emr-test-arn"
  release_label  = "emr-4.6.0"
  applications   = ["Spark"]

  termination_protection = false
  keep_job_flow_alive_when_no_steps = true

  ec2_attributes {
    subnet_id           = "${aws_subnet.main.id}"
    emr_managed_master_security_group = "${aws_security_group.sg.id}"
    emr_managed_slave_security_group  = "${aws_security_group.sg.id}"
    instance_profile       = "${aws_iam_instance_profile.emr_profile.arn}"
  }

  instance_group {
    instance_role = "CORE"
    instance_type = "c4.large"
    instance_count = "1"
    ebs_config {
      size = "40"
      type = "gp2"
      volumes_per_instance = 1
    }
    bid_price = "0.30"
    autoscaling_policy = <<EOF
{
  "Constraints": {
    "MinCapacity": 1,
    "MaxCapacity": 2
  },
  "Rules": [
    {
      "Name": "ScaleOutMemoryPercentage",
      "Description": "Scale out if YARNMemoryAvailablePercentage is less than 15",
      "Action": {
        "SimpleScalingPolicyConfiguration": {
          "AdjustmentType": "CHANGE_IN_CAPACITY",
          "ScalingAdjustment": 1,
          "CoolDown": 300
        }
      },
      "Trigger": {
```

```

        "CloudWatchAlarmDefinition": {
            "ComparisonOperator": "LESS_THAN",
            "EvaluationPeriods": 1,
            "MetricName": "YARNMemoryAvailablePercentage",
            "Namespace": "AWS/ElasticMapReduce",
            "Period": 300,
            "Statistic": "AVERAGE",
            "Threshold": 15.0,
            "Unit": "PERCENT"
        }
    }
}
]
}
EOF
}

ebs_root_volume_size      = 100

master_instance_type = "m3.xlarge"
core_instance_type   = "m3.xlarge"
core_instance_count   = 1

tags {
    role      = "rolename"
    env       = "env"
}

bootstrap_action {
    path = "s3://elasticmapreduce/bootstrap-actions/run-if"
    name = "runif"
    args = ["instance.isMaster=true", "echo running on master node"]
}

configurations = "test-fixtures/emr_configurations.json"

service_role = "${aws_iam_role.iam_emr_service_role.arn}"
}

```

The `aws_emr_cluster` resource typically requires two IAM roles, one for the EMR Cluster to use as a service, and another to place on your Cluster Instances to interact with AWS from those instances. The suggested role policy template for the EMR service is `AmazonElasticMapReduceRole`, and `AmazonElasticMapReduceforEC2Role` for the EC2 profile. See the Getting Started guide for more information on these IAM roles. There is also a fully-bootable example Terraform configuration at the bottom of this page.

## » Enable Debug Logging

Debug logging in EMR is implemented as a step. It is highly recommended to utilize the lifecycle configuration block with `ignore_changes` if other steps are being managed outside of Terraform.

```
resource "aws_emr_cluster" "example" {
  # ... other configuration ...

  step {
    action = "TERMINATE_CLUSTER"
    name   = "Setup Hadoop Debugging"

    hadoop_jar_step {
      jar = "command-runner.jar"
      args = ["state-pusher-script"]
    }
  }

  # Optional: ignore outside changes to running cluster steps
  lifecycle {
    ignore_changes = ["step"]
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the job flow
- **release\_label** - (Required) The release label for the Amazon EMR release
- **master\_instance\_type** - (Optional) The EC2 instance type of the master node. Exactly one of `master_instance_type` and `instance_group` must be specified.
- **scale\_down\_behavior** - (Optional) The way that individual Amazon EC2 instances terminate when an automatic scale-in activity occurs or an `instance_group` is resized.
- **service\_role** - (Required) IAM role that will be assumed by the Amazon EMR service to access AWS resources
- **security\_configuration** - (Optional) The security configuration name to attach to the EMR cluster. Only valid for EMR clusters with `release_label` 4.8.0 or greater
- **core\_instance\_type** - (Optional) The EC2 instance type of the slave nodes. Cannot be specified if `instance_groups` is set

- **core\_instance\_count** - (Optional) Number of Amazon EC2 instances used to execute the job flow. EMR will use one node as the cluster's master node and use the remainder of the nodes (**core\_instance\_count**-1) as core nodes. Cannot be specified if **instance\_groups** is set. Default 1
- **instance\_group** - (Optional) A list of **instance\_group** objects for each instance group in the cluster. Exactly one of **master\_instance\_type** and **instance\_group** must be specified. If **instance\_group** is set, then it must contain a configuration block for at least the **MASTER** instance group type (as well as any additional instance groups). Defined below
- **log\_uri** - (Optional) S3 bucket to write the log files of the job flow. If a value is not provided, logs are not created
- **applications** - (Optional) A list of applications for the cluster. Valid values are: **Flink**, **Hadoop**, **Hive**, **Mahout**, **Pig**, and **Spark**. Case insensitive
- **termination\_protection** - (Optional) Switch on/off termination protection (default is off)
- **keep\_job\_flow\_alive\_when\_no\_steps** - (Optional) Switch on/off run cluster with no steps or when all steps are complete (default is on)
- **ec2\_attributes** - (Optional) Attributes for the EC2 instances running the job flow. Defined below
- **kerberos\_attributes** - (Optional) Kerberos configuration for the cluster. Defined below
- **ebs\_root\_volume\_size** - (Optional) Size in GiB of the EBS root device volume of the Linux AMI that is used for each EC2 instance. Available in Amazon EMR version 4.x and later.
- **custom\_ami\_id** - (Optional) A custom Amazon Linux AMI for the cluster (instead of an EMR-owned AMI). Available in Amazon EMR version 5.7.0 and later.
- **bootstrap\_action** - (Optional) List of bootstrap actions that will be run before Hadoop is started on the cluster nodes. Defined below
- **configurations** - (Optional) List of configurations supplied for the EMR cluster you are creating
- **visible\_to\_all\_users** - (Optional) Whether the job flow is visible to all IAM users of the AWS account associated with the job flow. Default **true**
- **autoscaling\_role** - (Optional) An IAM role for automatic scaling policies. The IAM role provides permissions that the automatic scaling feature requires to launch and terminate EC2 instances in an instance group.
- **step** - (Optional) List of steps to run when creating the cluster. Defined below. It is highly recommended to utilize the lifecycle configuration block with **ignore\_changes** if other steps are being managed outside of Terraform.
- **tags** - (Optional) list of tags to apply to the EMR Cluster

## » `ec2__attributes`

Attributes for the Amazon EC2 instances running the job flow

- `key_name` - (Optional) Amazon EC2 key pair that can be used to ssh to the master node as the user called `hadoop`
- `subnet_id` - (Optional) VPC subnet id where you want the job flow to launch. Cannot specify the `cc1.4xlarge` instance type for nodes of a job flow launched in a Amazon VPC
- `additional_master_security_groups` - (Optional) String containing a comma separated list of additional Amazon EC2 security group IDs for the master node
- `additional_slave_security_groups` - (Optional) String containing a comma separated list of additional Amazon EC2 security group IDs for the slave nodes as a comma separated string
- `emr_managed_master_security_group` - (Optional) Identifier of the Amazon EC2 EMR-Managed security group for the master node
- `emr_managed_slave_security_group` - (Optional) Identifier of the Amazon EC2 EMR-Managed security group for the slave nodes
- `service_access_security_group` - (Optional) Identifier of the Amazon EC2 service-access security group - required when the cluster runs on a private subnet
- `instance_profile` - (Required) Instance Profile for EC2 instances of the cluster assume this role

**NOTE on EMR-Managed security groups:** These security groups will have any missing inbound or outbound access rules added and maintained by AWS, to ensure proper communication between instances in a cluster. The EMR service will maintain these rules for groups provided in `emr_managed_master_security_group` and `emr_managed_slave_security_group`; attempts to remove the required rules may succeed, only for the EMR service to re-add them in a matter of minutes. This may cause Terraform to fail to destroy an environment that contains an EMR cluster, because the EMR service does not revoke rules added on deletion, leaving a cyclic dependency between the security groups that prevents their deletion. To avoid this, use the `revoke_rules_on_delete` optional attribute for any Security Group used in `emr_managed_master_security_group` and `emr_managed_slave_security_group`. See Amazon EMR-Managed Security Groups for more information about the EMR-managed security group rules.

## » `kerberos__attributes`

Attributes for Kerberos configuration

- `ad_domain_join_password` - (Optional) The Active Directory password for `ad_domain_join_user`

- **ad\_domain\_join\_user** - (Optional) Required only when establishing a cross-realm trust with an Active Directory domain. A user with sufficient privileges to join resources to the domain.
- **cross\_realm\_trust\_principal\_password** - (Optional) Required only when establishing a cross-realm trust with a KDC in a different realm. The cross-realm principal password, which must be identical across realms.
- **kdc\_admin\_password** - (Required) The password used within the cluster for the kadmin service on the cluster-dedicated KDC, which maintains Kerberos principals, password policies, and keytabs for the cluster.
- **realm** - (Required) The name of the Kerberos realm to which all nodes in a cluster belong. For example, `EC2.INTERNAL`

## » **instance\_group**

Attributes for each task instance group in the cluster

- **instance\_role** - (Required) The role of the instance group in the cluster. Valid values are: `MASTER`, `CORE`, and `TASK`.
- **instance\_type** - (Required) The EC2 instance type for all instances in the instance group
- **instance\_count** - (Optional) Target number of instances for the instance group
- **name** - (Optional) Friendly name given to the instance group
- **bid\_price** - (Optional) If set, the bid price for each EC2 instance in the instance group, expressed in USD. By setting this attribute, the instance group is being declared as a Spot Instance, and will implicitly create a Spot request. Leave this blank to use On-Demand Instances. **bid\_price** can not be set for the `MASTER` instance group, since that group must always be On-Demand
- **ebs\_config** - (Optional) A list of attributes for the EBS volumes attached to each instance in the instance group. Each **ebs\_config** defined will result in additional EBS volumes being attached to *each* instance in the instance group. Defined below
- **autoscaling\_policy** - (Optional) The autoscaling policy document. This is a JSON formatted string. See EMR Auto Scaling

## » **ebs\_config**

Attributes for the EBS volumes attached to each EC2 instance in the **instance\_group**

- **size** - (Required) The volume size, in gibibytes (GiB).
- **type** - (Required) The volume type. Valid options are `gp2`, `io1`, and `standard`.



- **iops** - (Optional) The number of I/O operations per second (IOPS) that the volume supports
- **volumes\_per\_instance** - (Optional) The number of EBS volumes with this configuration to attach to each EC2 instance in the instance group (default is 1)

## » **bootstrap\_action**

- **name** - (Required) Name of the bootstrap action
- **path** - (Required) Location of the script to run during a bootstrap action. Can be either a location in Amazon S3 or on a local file system
- **args** - (Optional) List of command line arguments to pass to the bootstrap action script

## » **step**

Attributes for step configuration

- **action\_on\_failure** - (Required) The action to take if the step fails. Valid values: `TERMINATE_JOB_FLOW`, `TERMINATE_CLUSTER`, `CANCEL_AND_WAIT`, and `CONTINUE`
- **hadoop\_jar\_step** - (Required) The JAR file used for the step. Defined below.
- **name** - (Required) The name of the step.

## » **hadoop\_jar\_step**

Attributes for Hadoop job step configuration

- **args** - (Optional) List of command line arguments passed to the JAR file's main function when executed.
- **jar** - (Required) Path to a JAR file run during the step.
- **main\_class** - (Optional) Name of the main class in the specified Java file. If not specified, the JAR file should specify a Main-Class in its manifest file.
- **properties** - (Optional) Key-Value map of Java properties that are set when the step runs. You can use these properties to pass key value pairs to your main function.

## » **Attributes Reference**

The following attributes are exported:

- **id** - The ID of the EMR Cluster

- `name` - The name of the cluster.
- `release_label` - The release label for the Amazon EMR release.
- `master_instance_type` - The EC2 instance type of the master node.
- `master_public_dns` - The public DNS name of the master EC2 instance.
- `core_instance_type` - The EC2 instance type of the slave nodes.
- `core_instance_count` - The number of slave nodes, i.e. EC2 instance nodes.
- `log_uri` - The path to the Amazon S3 location where logs for this cluster are stored.
- `applications` - The applications installed on this cluster.
- `ec2_attributes` - Provides information about the EC2 instances in a cluster grouped by category: key name, subnet ID, IAM instance profile, and so on.
- `bootstrap_action` - A list of bootstrap actions that will be run before Hadoop is started on the cluster nodes.
- `configurations` - The list of Configurations supplied to the EMR cluster.
- `service_role` - The IAM role that will be assumed by the Amazon EMR service to access AWS resources on your behalf.
- `visible_to_all_users` - Indicates whether the job flow is visible to all IAM users of the AWS account associated with the job flow.
- `tags` - The list of tags associated with a cluster.

## » Example bootable config

**NOTE:** This configuration demonstrates a minimal configuration needed to boot an example EMR Cluster. It is not meant to display best practices. Please use at your own risk.

```
provider "aws" {
  region = "us-west-2"
}

resource "aws_emr_cluster" "tf-test-cluster" {
  name           = "emr-test-arn"
  release_label  = "emr-4.6.0"
  applications   = ["Spark"]

  ec2_attributes {
    subnet_id              = "${aws_subnet.main.id}"
    emr_managed_master_security_group = "${aws_security_group.allow_all.id}"
    emr_managed_slave_security_group  = "${aws_security_group.allow_all.id}"
    instance_profile          = "${aws_iam_instance_profile.emr_profile.arn}"
  }

  master_instance_type = "m3.xlarge"
```

```

core_instance_type    = "m3.xlarge"
core_instance_count   = 1

tags {
    role      = "rolename"
    dns_zone  = "env_zone"
    env       = "env"
    name      = "name-env"
}

bootstrap_action {
    path = "s3://elasticmapreduce/bootstrap-actions/run-if"
    name = "runif"
    args = ["instance.isMaster=true", "echo running on master node"]
}

configurations = "test-fixtures/emr_configurations.json"

service_role = "${aws_iam_role.iam_emr_service_role.arn}"
}

resource "aws_security_group" "allow_all" {
    name          = "allow_all"
    description   = "Allow all inbound traffic"
    vpc_id        = "${aws_vpc.main.id}"

    ingress {
        from_port    = 0
        to_port      = 0
        protocol      = "-1"
        cidr_blocks   = ["0.0.0.0/0"]
    }

    egress {
        from_port    = 0
        to_port      = 0
        protocol      = "-1"
        cidr_blocks   = ["0.0.0.0/0"]
    }
}

depends_on = ["aws_subnet.main"]

lifecycle {
    ignore_changes = ["ingress", "egress"]
}

```

```

    tags {
      name = "emr_test"
    }
  }

resource "aws_vpc" "main" {
  cidr_block      = "168.31.0.0/16"
  enable_dns_hostnames = true

  tags {
    name = "emr_test"
  }
}

resource "aws_subnet" "main" {
  vpc_id      = "${aws_vpc.main.id}"
  cidr_block = "168.31.0.0/20"

  tags {
    name = "emr_test"
  }
}

resource "aws_internet_gateway" "gw" {
  vpc_id = "${aws_vpc.main.id}"
}

resource "aws_route_table" "r" {
  vpc_id = "${aws_vpc.main.id}"

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = "${aws_internet_gateway.gw.id}"
  }
}

resource "aws_main_route_table_association" "a" {
  vpc_id      = "${aws_vpc.main.id}"
  route_table_id = "${aws_route_table.r.id}"
}

###

# IAM Role setups

###

```

```

# IAM role for EMR Service
resource "aws_iam_role" "iam_emr_service_role" {
  name = "iam_emr_service_role"

  assume_role_policy = <<EOF
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "elasticmapreduce.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF
}

resource "aws_iam_role_policy" "iam_emr_service_policy" {
  name = "iam_emr_service_policy"
  role = "${aws_iam_role.iam_emr_service_role.id}"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Resource": "*",
    "Action": [
      "ec2:AuthorizeSecurityGroupEgress",
      "ec2:AuthorizeSecurityGroupIngress",
      "ec2:CancelSpotInstanceRequests",
      "ec2:CreateNetworkInterface",
      "ec2:CreateSecurityGroup",
      "ec2:CreateTags",
      "ec2>DeleteNetworkInterface",
      "ec2>DeleteSecurityGroup",
      "ec2>DeleteTags",
      "ec2:DescribeAvailabilityZones",
      "ec2:DescribeAccountAttributes",
      "ec2:DescribeDhcpOptions",
      "ec2:DescribeInstanceStatus",

```

```

        "ec2:DescribeInstances",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeNetworkAcls",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribePrefixLists",
        "ec2:DescribeRouteTables",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeSpotInstanceRequests",
        "ec2:DescribeSpotPriceHistory",
        "ec2:DescribeSubnets",
        "ec2:DescribeVpcAttribute",
        "ec2:DescribeVpcEndpoints",
        "ec2:DescribeVpcEndpointServices",
        "ec2:DescribeVpcs",
        "ec2:DetachNetworkInterface",
        "ec2:ModifyImageAttribute",
        "ec2:ModifyInstanceAttribute",
        "ec2:RequestSpotInstances",
        "ec2:RevokeSecurityGroupEgress",
        "ec2:RunInstances",
        "ec2:TerminateInstances",
        "ec2:DeleteVolume",
        "ec2:DescribeVolumeStatus",
        "ec2:DescribeVolumes",
        "ec2:DetachVolume",
        "iam:GetRole",
        "iam:GetRolePolicy",
        "iam:ListInstanceProfiles",
        "iam:ListRolePolicies",
        "iam:PassRole",
        "s3:CreateBucket",
        "s3:Get*",
        "s3:List*",
        "sdb:BatchPutAttributes",
        "sdb:Select",
        "sqs:CreateQueue",
        "sqs:Delete*",
        "sqs:GetQueue*",
        "sqs:PurgeQueue",
        "sqs:ReceiveMessage"
    ]
}
EOF
}

```

```

# IAM Role for EC2 Instance Profile
resource "aws_iam_role" "iam_emr_profile_role" {
  name = "iam_emr_profile_role"

  assume_role_policy = <<EOF
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "ec2.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF
}

resource "aws_iam_instance_profile" "emr_profile" {
  name = "emr_profile"
  roles = ["${aws_iam_role.iam_emr_profile_role.name}"]
}

resource "aws_iam_role_policy" "iam_emr_profile_policy" {
  name = "iam_emr_profile_policy"
  role = "${aws_iam_role.iam_emr_profile_role.id}"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Resource": "*",
    "Action": [
      "cloudwatch:*",
      "dynamodb:*",
      "ec2:Describe*",
      "elasticmapreduce:Describe*",
      "elasticmapreduce:ListBootstrapActions",
      "elasticmapreduce:ListClusters",
      "elasticmapreduce:ListInstanceGroups",
      "elasticmapreduce:ListInstances",
      "elasticmapreduce:ListSteps",

```

```

        "kinesis:CreateStream",
        "kinesis>DeleteStream",
        "kinesis:DescribeStream",
        "kinesis:GetRecords",
        "kinesis:GetShardIterator",
        "kinesis:MergeShards",
        "kinesis:PutRecord",
        "kinesis:SplitShard",
        "rds:Describe*",
        "s3:*",
        "sdb:*",
        "sns:*",
        "sqs:*"
    ]
  }]
}
EOF
}
```

## » aws\_emr\_instance\_group

Provides an Elastic MapReduce Cluster Instance Group configuration. See Amazon Elastic MapReduce Documentation for more information.

**NOTE:** At this time, Instance Groups cannot be destroyed through the API nor web interface. Instance Groups are destroyed when the EMR Cluster is destroyed. Terraform will resize any Instance Group to zero when destroying the resource.

## » Example Usage

```

resource "aws_emr_instance_group" "task" {
  cluster_id      = "${aws_emr_cluster.tf-test-cluster.id}"
  instance_count  = 1
  instance_type   = "m3.xlarge"
  name            = "my little instance group"
}
```

## » Argument Reference

The following arguments are supported:



- **name** (Required) Human friendly name given to the instance group. Changing this forces a new resource to be created.
- **cluster\_id** (Required) ID of the EMR Cluster to attach to. Changing this forces a new resource to be created.
- **instance\_type** (Required) The EC2 instance type for all instances in the instance group. Changing this forces a new resource to be created.
- **instance\_count** (Optional) Target number of instances for the instance group. Defaults to 0.
- **ebs\_optimized** (Optional) Indicates whether an Amazon EBS volume is EBS-optimized. Changing this forces a new resource to be created.
- **ebs\_config** (Optional) One or more **ebs\_config** blocks as defined below. Changing this forces a new resource to be created.

**ebs\_config** supports the following:

- **iops** - (Optional) The number of I/O operations per second (IOPS) that the volume supports.
- **size** - (Optional) The volume size, in gibibytes (GiB). This can be a number from 1 - 1024. If the volume type is EBS-optimized, the minimum value is 10.
- **type** - (Optional) The volume type. Valid options are 'gp2', 'io1' and 'standard'.
- **volumes\_per\_instance** - (Optional) The number of EBS Volumes to attach per instance.

## » Attributes Reference

The following attributes are exported:

- **id** - The EMR Instance ID
- **running\_instance\_count** The number of instances currently running in this instance group.
- **status** The current status of the instance group.

## » **aws\_emr\_security\_configuration**

Provides a resource to manage AWS EMR Security Configurations

## » Example Usage

```
resource "aws_emr_security_configuration" "foo" {
  name = "emrsc_other"

  configuration = <<EOF
```

```

{
  "EncryptionConfiguration": {
    "AtRestEncryptionConfiguration": {
      "S3EncryptionConfiguration": {
        "EncryptionMode": "SSE-S3"
      },
      "LocalDiskEncryptionConfiguration": {
        "EncryptionKeyProviderType": "AwsKms",
        "AwsKmsKey": "arn:aws:kms:us-west-2:187416307283:alias/tf_emr_test_key"
      }
    },
    "EnableInTransitEncryption": false,
    "EnableAtRestEncryption": true
  }
}
EOF
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The name of the EMR Security Configuration. By default generated by Terraform.
- **name\_prefix** - (Optional) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **configuration** - (Required) A JSON formatted Security Configuration

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the EMR Security Configuration (Same as the **name**)
- **name** - The Name of the EMR Security Configuration
- **configuration** - The JSON formatted Security Configuration
- **creation\_date** - Date the Security Configuration was created

## » Import

EMR Security Configurations can be imported using the **name**, e.g.

```
$ terraform import aws_emr_security_configuraiton.sc example-sc-name
```

## » `aws_elasticsearch_domain`

### » Example Usage

```
resource "aws_elasticsearch_domain" "es" {
  domain_name      = "tf-test"
  elasticsearch_version = "1.5"
  cluster_config {
    instance_type = "r3.large.elasticsearch"
  }

  advanced_options {
    "rest.action.multi.allow_explicit_index" = "true"
  }

  access_policies = <<CONFIG
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "es:*",
      "Principal": "*",
      "Effect": "Allow",
      "Condition": {
        "IpAddress": {"aws:SourceIp": ["66.193.100.22/32"]}
      }
    }
  ]
}
CONFIG

  snapshot_options {
    automated_snapshot_start_hour = 23
  }

  tags {
    Domain = "TestDomain"
  }
}
```

### » Argument Reference

The following arguments are supported:

- `domain_name` - (Required) Name of the domain.

- `access_policies` - (Optional) IAM policy document specifying the access policies for the domain
- `advanced_options` - (Optional) Key-value string pairs to specify advanced configuration options.
- `ebs_options` - (Optional) EBS related options, may be required based on chosen instance size. See below.
- `encrypt_at_rest` - (Optional) Encrypt at rest options. Only available for certain instance types. See below.
- `cluster_config` - (Optional) Cluster configuration of the domain, see below.
- `snapshot_options` - (Optional) Snapshot related options, see below.
- `vpc_options` - (Optional) VPC related options, see below. Adding or removing this configuration forces a new resource (documentation).
- `log_publishing_options` - (Optional) Options for publishing slow logs to CloudWatch Logs.
- `elasticsearch_version` - (Optional) The version of Elasticsearch to deploy. Defaults to 1.5
- `tags` - (Optional) A mapping of tags to assign to the resource

`ebs_options` supports the following attributes:

- `ebs_enabled` - (Required) Whether EBS volumes are attached to data nodes in the domain
- `volume_type` - (Optional) The type of EBS volumes attached to data nodes.
- `volume_size` - The size of EBS volumes attached to data nodes (in GB). **Required** if `ebs_enabled` is set to `true`.
- `iops` - (Optional) The baseline input/output (I/O) performance of EBS volumes attached to data nodes. Applicable only for the Provisioned IOPS EBS volume type.

`encrypt_at_rest` supports the following attributes:

- `enabled` - (Required) Whether to enable encryption at rest. If the `encrypt_at_rest` block is not provided then this defaults to `false`.
- `kms_key_id` - (Optional) The KMS key id to encrypt the Elasticsearch domain with. If not specified then it defaults to using the `aws/es` service KMS key.

`cluster_config` supports the following attributes:

- `instance_type` - (Optional) Instance type of data nodes in the cluster.
- `instance_count` - (Optional) Number of instances in the cluster.
- `dedicated_master_enabled` - (Optional) Indicates whether dedicated master nodes are enabled for the cluster.
- `dedicated_master_type` - (Optional) Instance type of the dedicated master nodes in the cluster.
- `dedicated_master_count` - (Optional) Number of dedicated master nodes in the cluster

- **zone\_awareness\_enabled** - (Optional) Indicates whether zone awareness is enabled.

**vpc\_options** supports the following attributes:

AWS documentation: [VPC Support for Amazon Elasticsearch Service Domains](#)

- **security\_group\_ids** - (Optional) List of VPC Security Group IDs to be applied to the Elasticsearch domain endpoints. If omitted, the default Security Group for the VPC will be used.
- **subnet\_ids** - (Required) List of VPC Subnet IDs for the Elasticsearch domain endpoints to be created in.

Security Groups and Subnets referenced in these attributes must all be within the same VPC; this determines what VPC the endpoints are created in.

**snapshot\_options** supports the following attribute:

- **automated\_snapshot\_start\_hour** - (Required) Hour during which the service takes an automated daily snapshot of the indices in the domain.

**log\_publishing\_options** supports the following attribute:

- **log\_type** - (Required) A type of Elasticsearch log. Valid values: INDEX\_SLOW\_LOGS, SEARCH\_SLOW\_LOGS
- **cloudwatch\_log\_group\_arn** - (Required) ARN of the Cloudwatch log group to which log needs to be published.
- **enabled** - (Optional, Default: true) Specifies whether given log publishing option is enabled or not.

## » Attributes Reference

The following attributes are exported:

- **arn** - Amazon Resource Name (ARN) of the domain.
- **domain\_id** - Unique identifier for the domain.
- **endpoint** - Domain-specific endpoint used to submit index, search, and data upload requests.
- **kibana\_endpoint** - Domain-specific endpoint for kibana without https scheme.
- **vpc\_options.0.availability\_zones** - If the domain was created inside a VPC, the names of the availability zones the configured **subnet\_ids** were created inside.
- **vpc\_options.0.vpc\_id** - If the domain was created inside a VPC, the ID of the VPC.

## » Import

ElasticSearch domains can be imported using the `domain_name`, e.g.

```
$ terraform import aws_elasticsearch_domain.example domain_name
```

## » `aws_elasticsearch_domain_policy`

Allows setting policy to an ElasticSearch domain while referencing domain attributes (e.g. ARN)

## » Example Usage

```
resource "aws_elasticsearch_domain" "example" {
  domain_name      = "tf-test"
  elasticsearch_version = "2.3"
}

resource "aws_elasticsearch_domain_policy" "main" {
  domain_name = "${aws_elasticsearch_domain.example.domain_name}"

  access_policies = <<POLICIES
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "es:*",
      "Principal": "*",
      "Effect": "Allow",
      "Condition": {
        "IpAddress": {"aws:SourceIp": "127.0.0.1/32"}
      },
      "Resource": "${aws_elasticsearch_domain.example.arn}"
    }
  ]
}
POLICIES
}
```

## » Argument Reference

The following arguments are supported:

- **domain\_name** - (Required) Name of the domain.
- **access\_policies** - (Optional) IAM policy document specifying the access policies for the domain

## » **aws\_\_elastictranscoder\_\_pipeline**

Provides an Elastic Transcoder pipeline resource.

### » **Example Usage**

```
resource "aws_elastictranscoder_pipeline" "bar" {
  input_bucket = "${aws_s3_bucket.input_bucket.bucket}"
  name         = "aws_elastictranscoder_pipeline_tf_test_"
  role         = "${aws_iam_role.test_role.arn}"

  content_config = {
    bucket      = "${aws_s3_bucket.content_bucket.bucket}"
    storage_class = "Standard"
  }

  thumbnail_config = {
    bucket      = "${aws_s3_bucket.thumb_bucket.bucket}"
    storage_class = "Standard"
  }
}
```

### » **Argument Reference**

See "Create Pipeline" in the AWS docs for reference.

The following arguments are supported:

- **aws\_kms\_key\_arn** - (Optional) The AWS Key Management Service (AWS KMS) key that you want to use with this pipeline.
- **content\_config** - (Optional) The ContentConfig object specifies information about the Amazon S3 bucket in which you want Elastic Transcoder to save transcoded files and playlists. (documented below)
- **content\_config\_permissions** - (Optional) The permissions for the **content\_config** object. (documented below)
- **input\_bucket** - (Required) The Amazon S3 bucket in which you saved the media files that you want to transcode and the graphics that you want to use as watermarks.

- **name** - (Optional, Forces new resource) The name of the pipeline. Maximum 40 characters
- **notifications** - (Optional) The Amazon Simple Notification Service (Amazon SNS) topic that you want to notify to report job status. (documented below)
- **output\_bucket** - (Optional) The Amazon S3 bucket in which you want Elastic Transcoder to save the transcoded files.
- **role** - (Required) The IAM Amazon Resource Name (ARN) for the role that you want Elastic Transcoder to use to transcode jobs for this pipeline.
- **thumbnail\_config** - (Optional) The ThumbnailConfig object specifies information about the Amazon S3 bucket in which you want Elastic Transcoder to save thumbnail files. (documented below)
- **thumbnail\_config\_permissions** - (Optional) The permissions for the thumbnail\_config object. (documented below)

The **content\_config** object specifies information about the Amazon S3 bucket in which you want Elastic Transcoder to save transcoded files and playlists: which bucket to use, and the storage class that you want to assign to the files. If you specify values for **content\_config**, you must also specify values for **thumbnail\_config**. If you specify values for **content\_config** and **thumbnail\_config**, omit the **output\_bucket** object.

The **content\_config** object supports the following:

- **bucket** - The Amazon S3 bucket in which you want Elastic Transcoder to save transcoded files and playlists.
- **storage\_class** - The Amazon S3 storage class, Standard or ReducedRedundancy, that you want Elastic Transcoder to assign to the files and playlists that it stores in your Amazon S3 bucket.

The **content\_config\_permissions** object supports the following:

- **access** - The permission that you want to give to the AWS user that you specified in **content\_config\_permissions.grantee**
- **grantee** - The AWS user or group that you want to have access to transcoded files and playlists.
- **grantee\_type** - Specify the type of value that appears in the **content\_config\_permissions.grantee** object. Valid values are Canonical, Email or Group.

The **notifications** object supports the following:

- **completed** - The topic ARN for the Amazon SNS topic that you want to notify when Elastic Transcoder has finished processing a job in this pipeline.
- **error** - The topic ARN for the Amazon SNS topic that you want to notify when Elastic Transcoder encounters an error condition while processing a job in this pipeline.
- **progressing** - The topic ARN for the Amazon Simple Notification Service



(Amazon SNS) topic that you want to notify when Elastic Transcoder has started to process a job in this pipeline.

- **warning** - The topic ARN for the Amazon SNS topic that you want to notify when Elastic Transcoder encounters a warning condition while processing a job in this pipeline.

The `thumbnail_config` object specifies information about the Amazon S3 bucket in which you want Elastic Transcoder to save thumbnail files: which bucket to use, which users you want to have access to the files, the type of access you want users to have, and the storage class that you want to assign to the files. If you specify values for `content_config`, you must also specify values for `thumbnail_config` even if you don't want to create thumbnails. (You control whether to create thumbnails when you create a job. For more information, see `ThumbnailPattern` in the topic `Create Job`.) If you specify values for `content_config` and `thumbnail_config`, omit the `OutputBucket` object.

The `thumbnail_config` object supports the following:

- **bucket** - The Amazon S3 bucket in which you want Elastic Transcoder to save thumbnail files.
- **storage\_class** - The Amazon S3 storage class, `Standard` or `ReducedRedundancy`, that you want Elastic Transcoder to assign to the thumbnails that it stores in your Amazon S3 bucket.

The `thumbnail_config_permissions` object supports the following:

- **access** - The permission that you want to give to the AWS user that you specified in `thumbnail_config_permissions.grantee`.
- **grantee** - The AWS user or group that you want to have access to thumbnail files.
- **grantee\_type** - Specify the type of value that appears in the `thumbnail_config_permissions.grantee` object.

## » `aws_elastictranscoder_preset`

Provides an Elastic Transcoder preset resource.

### » Example Usage

```
resource "aws_elastictranscoder_preset" "bar" {
  container    = "mp4"
  description  = "Sample Preset"
  name         = "sample_preset"

  audio = {
```

```

    audio_packing_mode = "SingleTrack"
    bit_rate            = 96
    channels            = 2
    codec               = "AAC"
    sample_rate         = 44100
}

audio_codec_options = {
  profile = "AAC-LC"
}

video = {
  bit_rate            = "1600"
  codec              = "H.264"
  display_aspect_ratio = "16:9"
  fixed_gop          = "false"
  frame_rate         = "auto"
  max_frame_rate     = "60"
  keyframes_max_dist = 240
  max_height         = "auto"
  max_width          = "auto"
  padding_policy     = "Pad"
  sizing_policy      = "Fit"
}

video_codec_options = {
  Profile              = "main"
  Level               = "2.2"
  MaxReferenceFrames  = 3
  InterlaceMode       = "Progressive"
  ColorSpaceConversionMode = "None"
}

video_watermarks = {
  id                  = "Terraform Test"
  max_width           = "20%"
  max_height          = "20%"
  sizing_policy       = "ShrinkToFit"
  horizontal_align    = "Right"
  horizontal_offset   = "10px"
  vertical_align      = "Bottom"
  vertical_offset     = "10px"
  opacity             = "55.5"
  target              = "Content"
}

```

```

    thumbnails = {
        format      = "png"
        interval    = 120
        max_width   = "auto"
        max_height  = "auto"
        padding_policy = "Pad"
        sizing_policy = "Fit"
    }
}

```

## » Argument Reference

See "Create Preset" in the AWS docs for reference.

The following arguments are supported:

- **audio** - (Optional, Forces new resource) Audio parameters object (documented below).
- **audio\_codec\_options** - (Optional, Forces new resource) Codec options for the audio parameters (documented below)
- **container** - (Required, Forces new resource) The container type for the output file. Valid values are `flac`, `flv`, `fmp4`, `gif`, `mp3`, `mp4`, `mpg`, `mxp`, `oga`, `ogg`, `ts`, and `webm`.
- **description** - (Optional, Forces new resource) A description of the preset (maximum 255 characters)
- **name** - (Optional, Forces new resource) The name of the preset. (maximum 40 characters)
- **thumbnails** - (Optional, Forces new resource) Thumbnail parameters object (documented below)
- **video** - (Optional, Forces new resource) Video parameters object (documented below)
- **video\_watermarks** - (Optional, Forces new resource) Watermark parameters for the video parameters (documented below)
- **video\_codec\_options** (Optional, Forces new resource) Codec options for the video parameters

The **audio** object supports the following:

- **audio\_packing\_mode** - The method of organizing audio channels and tracks. Use `Audio:Channels` to specify the number of channels in your output, and `Audio:AudioPackingMode` to specify the number of tracks and their relation to the channels. If you do not specify an `Audio:AudioPackingMode`, Elastic Transcoder uses `SingleTrack`.
- **bit\_rate** - The bit rate of the audio stream in the output file, in kilobits/second. Enter an integer between 64 and 320, inclusive.
- **channels** - The number of audio channels in the output file

- **codec** - The audio codec for the output file. Valid values are **AAC**, **flac**, **mp2**, **mp3**, **pcm**, and **vorbis**.
- **sample\_rate** - The sample rate of the audio stream in the output file, in hertz. Valid values are: **auto**, 22050, 32000, 44100, 48000, 96000

The **audio\_codec\_options** object supports the following:

- **bit\_depth** - The bit depth of a sample is how many bits of information are included in the audio samples. Valid values are 16 and 24. (FLAC/PCM Only)
- **bit\_order** - The order the bits of a PCM sample are stored in. The supported value is LittleEndian. (PCM Only)
- **profile** - If you specified AAC for Audio:Codec, choose the AAC profile for the output file.
- **signed** - Whether audio samples are represented with negative and positive numbers (signed) or only positive numbers (unsigned). The supported value is Signed. (PCM Only)

The **thumbnails** object supports the following:

- **aspect\_ratio** - The aspect ratio of thumbnails. The following values are valid: **auto**, 1:1, 4:3, 3:2, 16:9
- **format** - The format of thumbnails, if any. Valid formats are **jpg** and **png**.
- **interval** - The approximate number of seconds between thumbnails. The value must be an integer. The actual interval can vary by several seconds from one thumbnail to the next.
- **max\_height** - The maximum height of thumbnails, in pixels. If you specify **auto**, Elastic Transcoder uses 1080 (Full HD) as the default value. If you specify a numeric value, enter an even integer between 32 and 3072, inclusive.
- **max\_width** - The maximum width of thumbnails, in pixels. If you specify **auto**, Elastic Transcoder uses 1920 (Full HD) as the default value. If you specify a numeric value, enter an even integer between 32 and 4096, inclusive.
- **padding\_policy** - When you set **PaddingPolicy** to **Pad**, Elastic Transcoder might add black bars to the top and bottom and/or left and right sides of thumbnails to make the total size of the thumbnails match the values that you specified for thumbnail **MaxWidth** and **MaxHeight** settings.
- **resolution** - The width and height of thumbnail files in pixels, in the format **WidthxHeight**, where both values are even integers. The values cannot exceed the width and height that you specified in the **Video:Resolution** object. (To better control resolution and aspect ratio of thumbnails, we recommend that you use the thumbnail values **max\_width**, **max\_height**, **sizing\_policy**, and **padding\_policy** instead of **resolution** and **aspect\_ratio**. The two groups of settings are mutually exclusive. Do not use them together)

- **sizing\_policy** - A value that controls scaling of thumbnails. Valid values are: **Fit**, **Fill**, **Stretch**, **Keep**, **ShrinkToFit**, and **ShrinkToFill**.

The **video** object supports the following:

- **aspect\_ratio** - The display aspect ratio of the video in the output file. Valid values are: **auto**, **1:1**, **4:3**, **3:2**, **16:9**. (Note; to better control resolution and aspect ratio of output videos, we recommend that you use the values **max\_width**, **max\_height**, **sizing\_policy**, **padding\_policy**, and **display\_aspect\_ratio** instead of **resolution** and **aspect\_ratio**.)
- **bit\_rate** - The bit rate of the video stream in the output file, in kilo-bits/second. You can configure variable bit rate or constant bit rate encoding.
- **codec** - The video codec for the output file. Valid values are **gif**, **H.264**, **mpeg2**, **vp8**, and **vp9**.
- **display\_aspect\_ratio** - The value that Elastic Transcoder adds to the metadata in the output file. If you set **DisplayAspectRatio** to **auto**, Elastic Transcoder chooses an aspect ratio that ensures square pixels. If you specify another option, Elastic Transcoder sets that value in the output file.
- **fixed\_gop** - Whether to use a fixed value for **Video:FixedGOP**. Not applicable for containers of type **gif**. Valid values are **true** and **false**.
- **frame\_rate** - The frames per second for the video stream in the output file. The following values are valid: **auto**, **10**, **15**, **23.97**, **24**, **25**, **29.97**, **30**, **50**, **60**.
- **keyframes\_max\_dist** - The maximum number of frames between key frames. Not applicable for containers of type **gif**.
- **max\_frame\_rate** - If you specify **auto** for **FrameRate**, Elastic Transcoder uses the frame rate of the input video for the frame rate of the output video, up to the maximum frame rate. If you do not specify a **MaxFrameRate**, Elastic Transcoder will use a default of **30**.
- **max\_height** - The maximum height of the output video in pixels. If you specify **auto**, Elastic Transcoder uses **1080** (Full HD) as the default value. If you specify a numeric value, enter an even integer between **96** and **3072**, inclusive.
- **max\_width** - The maximum width of the output video in pixels. If you specify **auto**, Elastic Transcoder uses **1920** (Full HD) as the default value. If you specify a numeric value, enter an even integer between **128** and **4096**, inclusive.
- **padding\_policy** - When you set **PaddingPolicy** to **Pad**, Elastic Transcoder might add black bars to the top and bottom and/or left and right sides of the output video to make the total size of the output video match the values that you specified for **max\_width** and **max\_height**.
- **resolution** - The width and height of the video in the output file, in pixels. Valid values are **auto** and **widthxheight**. (see note for **aspect\_ratio**)
- **sizing\_policy** - A value that controls scaling of the output video. Valid values are: **Fit**, **Fill**, **Stretch**, **Keep**, **ShrinkToFit**, **ShrinkToFill**.

The `video_watermarks` object supports the following:

- `horizontal_align` - The horizontal position of the watermark unless you specify a nonzero value for `horizontal_offset`.
- `horizontal_offset` - The amount by which you want the horizontal position of the watermark to be offset from the position specified by `horizontal_align`.
- `id` - A unique identifier for the settings for one watermark. The value of `id` can be up to 40 characters long. You can specify settings for up to four watermarks.
- `max_height` - The maximum height of the watermark.
- `max_width` - The maximum width of the watermark.
- `opacity` - A percentage that indicates how much you want a watermark to obscure the video in the location where it appears.
- `sizing_policy` - A value that controls scaling of the watermark. Valid values are: `Fit`, `Stretch`, `ShrinkToFit`
- `target` - A value that determines how Elastic Transcoder interprets values that you specified for `video_watermarks.horizontal_offset`, `video_watermarks.vertical_offset`, `video_watermarks.max_width`, and `video_watermarks.max_height`. Valid values are `Content` and `Frame`.
- `vertical_align` - The vertical position of the watermark unless you specify a nonzero value for `vertical_align`. Valid values are `Top`, `Bottom`, `Center`.
- `vertical_offset` - The amount by which you want the vertical position of the watermark to be offset from the position specified by `vertical_align`

The `video_codec_options` map supports the following:

- `Profile` - The codec profile that you want to use for the output file. (H.264/VP8 Only)
- `Level` - The H.264 level that you want to use for the output file. Elastic Transcoder supports the following levels: `1`, `1b`, `1.1`, `1.2`, `1.3`, `2`, `2.1`, `2.2`, `3`, `3.1`, `3.2`, `4`, `4.1` (H.264 only)
- `MaxReferenceFrames` - The maximum number of previously decoded frames to use as a reference for decoding future frames. Valid values are integers 0 through 16. (H.264 only)
- `MaxBitRate` - The maximum number of kilobits per second in the output video. Specify a value between 16 and 62,500 inclusive, or `auto`. (Optional, H.264/MPEG2/VP8/VP9 only)
- `BufferSize` - The maximum number of kilobits in any x seconds of the output video. This window is commonly 10 seconds, the standard segment duration when you're using `ts` for the container type of the output video. Specify an integer greater than 0. If you specify `MaxBitRate` and omit `BufferSize`, Elastic Transcoder sets `BufferSize` to 10 times the value of `MaxBitRate`. (Optional, H.264/MPEG2/VP8/VP9 only)
- `InterlacedMode` - The interlace mode for the output video. (Optional,

H.264/MPEG2 Only)

- **ColorSpaceConversion** - The color space conversion Elastic Transcoder applies to the output video. Valid values are **None**, **Bt709toBt601**, **Bt601toBt709**, and **Auto**. (Optional, H.264/MPEG2 Only)
- **ChromaSubsampling** - The sampling pattern for the chroma (color) channels of the output video. Valid values are **yuv420p** and **yuv422p**.
- **LoopCount** - The number of times you want the output gif to loop (Gif only)

## » **aws\_gamelift\_alias**

Provides a Gamelift Alias resource.

### » **Example Usage**

```
resource "aws_gamelift_alias" "example" {
  name = "example-alias"
  description = "Example Description"
  routing_strategy {
    message = "Example Message"
    type = "TERMINAL"
  }
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) Name of the alias.
- **description** - (Optional) Description of the alias.
- **routing\_strategy** - (Required) Specifies the fleet and/or routing type to use for the alias.

### » **Nested Fields**

#### » **routing\_strategy**

- **fleet\_id** - (Optional) ID of the Gamelift Fleet to point the alias to.
- **message** - (Optional) Message text to be used with the **TERMINAL** routing strategy.
- **type** - (Required) Type of routing strategy. e.g. **SIMPLE** or **TERMINAL**

## » Attributes Reference

The following attributes are exported:

- `id` - Alias ID.
- `arn` - Alias ARN.

## » Import

Gamelift Aliases can be imported using the ID, e.g.

```
$ terraform import aws_gamelift_alias.example <alias-id>
```

## » `aws_gamelift_build`

Provides an Gamelift Build resource.

## » Example Usage

```
resource "aws_gamelift_build" "test" {
  name = "example-build"
  operating_system = "WINDOWS_2012"
  storage_location {
    bucket = "${aws_s3_bucket.test.bucket}"
    key = "${aws_s3_bucket_object.test.key}"
    role_arn = "${aws_iam_role.test.arn}"
  }
  depends_on = ["aws_iam_role_policy.test"]
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the build
- `operating_system` - (Required) Operating system that the game server binaries are built to run on. e.g. `WINDOWS_2012` or `AMAZON_LINUX`.
- `storage_location` - (Required) Information indicating where your game build files are stored. See below.
- `version` - (Optional) Version that is associated with this build.



## » Nested Fields

### » `storage_location`

- `bucket` - (Required) Name of your S3 bucket.
- `key` - (Required) Name of the zip file containing your build files.
- `role_arn` - (Required) ARN of the access role that allows Amazon GameLift to access your S3 bucket.

## » Attributes Reference

The following attributes are exported:

- `id` - Build ID.

## » Import

GameLift Builds cannot be imported at this time.

## » `aws_gamelift_fleet`

Provides a GameLift Fleet resource.

## » Example Usage

```
resource "aws_gamelift_fleet" "example" {
  build_id = "${aws_gamelift_build.example.id}"
  ec2_instance_type = "t2.micro"
  name = "example-fleet-name"
  runtime_configuration {
    server_process {
      concurrent_executions = 1
      launch_path = "C:\\game\\GomokuServer.exe"
    }
  }
}
```

## » Argument Reference

The following arguments are supported:

- **build\_id** - (Required) ID of the Gamelift Build to be deployed on the fleet.
- **ec2\_instance\_type** - (Required) Name of an EC2 instance type. e.g. `t2.micro`
- **name** - (Required) The name of the fleet.
- **description** - (Optional) Human-readable description of the fleet.
- **ec2\_inbound\_permission** - (Optional) Range of IP addresses and port settings that permit inbound traffic to access server processes running on the fleet. See below.
- **metric\_groups** - (Optional) List of names of metric groups to add this fleet to. A metric group tracks metrics across all fleets in the group. Defaults to `default`.
- **new\_game\_session\_protection\_policy** - (Optional) Game session protection policy to apply to all instances in this fleet. e.g. `FullProtection`. Defaults to `NoProtection`.
- **resource\_creation\_limit\_policy** - (Optional) Policy that limits the number of game sessions an individual player can create over a span of time for this fleet. See below.
- **runtime\_configuration** - (Optional) Instructions for launching server processes on each instance in the fleet. See below.

## » Nested Fields

### » **ec2\_inbound\_permission**

- **from\_port** - (Required) Starting value for a range of allowed port numbers.
- **ip\_range** - (Required) Range of allowed IP addresses expressed in CIDR notation. e.g. `000.000.000.000/[subnet mask]` or `0.0.0.0/[subnet mask]`.
- **protocol** - (Required) Network communication protocol used by the fleet. e.g. `TCP` or `UDP`
- **to\_port** - (Required) Ending value for a range of allowed port numbers. Port numbers are end-inclusive. This value must be higher than **from\_port**.

### » **resource\_creation\_limit\_policy**

- **new\_game\_sessions\_per\_creator** - (Optional) Maximum number of game sessions that an individual can create during the policy period.
- **policy\_period\_in\_minutes** - (Optional) Time span used in evaluating the resource creation limit policy.

### » **runtime\_configuration**

- `game_session_activation_timeout_seconds` - (Optional) Maximum amount of time (in seconds) that a game session can remain in status `ACTIVATING`.
- `max_concurrent_game_session_activations` - (Optional) Maximum number of game sessions with status `ACTIVATING` to allow on an instance simultaneously.
- `server_process` - (Optional) Collection of server process configurations that describe which server processes to run on each instance in a fleet. See below.

#### » `server_process`

- `concurrent_executions` - (Required) Number of server processes using this configuration to run concurrently on an instance.
- `launch_path` - (Required) Location of the server executable in a game build. All game builds are installed on instances at the root : for Windows instances `C:\game`, and for Linux instances `/local/game`.
- `parameters` - (Optional) Optional list of parameters to pass to the server executable on launch.

### » Attributes Reference

The following attributes are exported:

- `id` - Fleet ID.
- `arn` - Fleet ARN.
- `operating_system` - Operating system of the fleet's computing resources.

### » Import

Gamelift Fleets cannot be imported at this time.

### » `aws_glacier_vault`

Provides a Glacier Vault Resource. You can refer to the Glacier Developer Guide for a full explanation of the Glacier Vault functionality

**NOTE:** When removing a Glacier Vault, the Vault must be empty.

### » Example Usage

```
resource "aws_sns_topic" "aws_sns_topic" {
```

```

    name = "glacier-sns-topic"
}

resource "aws_glacier_vault" "my_archive" {
    name = "MyArchive"

    notification {
        sns_topic = "${aws_sns_topic.aws_sns_topic.arn}"
        events    = ["ArchiveRetrievalCompleted", "InventoryRetrievalCompleted"]
    }

    access_policy = <<EOF
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "add-read-only-perm",
            "Principal": "*",
            "Effect": "Allow",
            "Action": [
                "glacier:InitiateJob",
                "glacier:GetJobOutput"
            ],
            "Resource": "arn:aws:glacier:eu-west-1:432981146916:vaults/MyArchive"
        }
    ]
}
    ]
}
EOF

    tags {
        Test = "MyArchive"
    }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the Vault. Names can be between 1 and 255 characters long and the valid characters are a-z, A-Z, 0-9, '\_' (underscore), '-' (hyphen), and '.' (period).
- **access\_policy** - (Optional) The policy document. This is a JSON formatted string. The heredoc syntax or `file` function is helpful here. Use the Glacier Developer Guide for more information on Glacier Vault Policy
- **notification** - (Optional) The notifications for the Vault. Fields docu-

mented below.

- **tags** - (Optional) A mapping of tags to assign to the resource.

**notification** supports the following:

- **events** - (Required) You can configure a vault to publish a notification for `ArchiveRetrievalCompleted` and `InventoryRetrievalCompleted` events.
- **sns\_topic** - (Required) The SNS Topic ARN.

## » Attributes Reference

The following attributes are exported:

- **location** - The URI of the vault that was created.
- **arn** - The ARN of the vault.

## » Import

Glacier Vaults can be imported using the **name**, e.g.

```
$ terraform import aws_glacier_vault.archive my_archive
```

## » aws\_glue\_catalog\_database

Provides a Glue Catalog Database Resource. You can refer to the Glue Developer Guide for a full explanation of the Glue Data Catalog functionality

## » Example Usage

```
resource "aws_glue_catalog_database" "aws_glue_catalog_database" {  
  name = "MyCatalogDatabase"  
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the database.
- **catalog\_id** - (Optional) ID of the Glue Catalog to create the database in. If omitted, this defaults to the AWS Account ID.
- **description** - (Optional) Description of the database.

- `location_uri` - (Optional) The location of the database (for example, an HDFS path).
- `parameters` - (Optional) A list of key-value pairs that define parameters and properties of the database.

## » Import

Glue Catalog Databases can be imported using the `catalog_id:name`. If you have not set a Catalog ID specify the AWS Account ID that the database is in, e.g.

```
$ terraform import aws_glue_catalog_database.database 123456789012:my_database
```

## » `aws_glue_connection`

Provides a Glue Connection resource.

### » Example Usage

#### » Non-VPC Connection

```
resource "aws_glue_connection" "example" {
  connection_properties = {
    JDBC_CONNECTION_URL = "jdbc:mysql://example.com/exampledatabase"
    PASSWORD             = "examplepassword"
    USERNAME             = "exampleusername"
  }

  name = "example"
}
```

#### » VPC Connection

For more information, see the AWS Documentation.

```
resource "aws_glue_connection" "example" {
  connection_properties = {
    JDBC_CONNECTION_URL = "jdbc:mysql://${aws_rds_cluster.example.endpoint}/exampledatabase"
    PASSWORD             = "examplepassword"
    USERNAME             = "exampleusername"
  }

  name = "example"
```

```

physical_connection_requirements {
  security_group_id_list = ["${aws_security_group.example.id}"]
  subnet_id              = "${aws_subnet.example.id}"
}

```

## » Argument Reference

The following arguments are supported:

- **catalog\_id** – (Optional) The ID of the Data Catalog in which to create the connection. If none is supplied, the AWS account ID is used by default.
- **connection\_properties** – (Required) A map of key-value pairs used as parameters for this connection.
- **connection\_type** – (Optional) The type of the connection. Defaults to JDBC.
- **description** – (Optional) Description of the connection.
- **match\_criteria** – (Optional) A list of criteria that can be used in selecting this connection.
- **name** – (Required) The name of the connection.
- **physical\_connection\_requirements** – (Optional) A map of physical connection requirements, such as VPC and SecurityGroup. Defined below.

### » **physical\_connection\_requirements**

- **security\_group\_id\_list** – (Optional) The security group ID list used by the connection.
- **subnet\_id** – (Optional) The subnet ID used by the connection.

## » Attributes Reference

The following additional attributes are exported:

- **id** - Catalog ID and name of the connection

## » Import

Glue Connections can be imported using the **CATALOG-ID** (AWS account ID if not custom) and **NAME**, e.g.

```
$ terraform import aws_glue_connection.MyConnection 123456789012:MyConnection
```

## » `aws_glue_job`

Provides a Glue Job resource.

### » Example Usage

#### » Python Job

```
resource "aws_glue_job" "example" {
  name      = "example"
  role_arn = "${aws_iam_role.example.arn}"

  command {
    script_location = "s3://${aws_s3_bucket.example.bucket}/example.py"
  }
}
```

#### » Scala Job

```
resource "aws_glue_job" "example" {
  name      = "example"
  role_arn = "${aws_iam_role.example.arn}"

  command {
    script_location = "s3://${aws_s3_bucket.example.bucket}/example.scala"
  }

  default_arguments = {
    "--job-language" = "scala"
  }
}
```

### » Argument Reference

The following arguments are supported:

- `allocated_capacity` – (Optional) The number of AWS Glue data processing units (DPUs) to allocate to this Job. From 2 to 100 DPUs can be allocated; the default is 10. A DPU is a relative measure of processing power that consists of 4 vCPUs of compute capacity and 16 GB of memory.
- `command` – (Required) The command of the job. Defined below.
- `connections` – (Optional) The list of connections used for this job.



- **default\_arguments** – (Optional) The map of default arguments for this job. You can specify arguments here that your own job-execution script consumes, as well as arguments that AWS Glue itself consumes. For information about how to specify and consume your own Job arguments, see the Calling AWS Glue APIs in Python topic in the developer guide. For information about the key-value pairs that AWS Glue consumes to set up your job, see the Special Parameters Used by AWS Glue topic in the developer guide.
- **description** – (Optional) Description of the job.
- **execution\_property** – (Optional) Execution property of the job. Defined below.
- **max\_retries** – (Optional) The maximum number of times to retry this job if it fails.
- **name** – (Required) The name you assign to this job. It must be unique in your account.
- **role** – (Required) The ARN of the IAM role associated with this job.

#### » **command Argument Reference**

- **name** - (Optional) The name of the job command. Defaults to `glueetl`
- **script\_location** - (Required) Specifies the S3 path to a script that executes a job.

#### » **execution\_property Argument Reference**

- **max\_concurrent\_runs** - (Optional) The maximum number of concurrent runs allowed for a job. The default is 1.

#### » **Attributes Reference**

The following additional attributes are exported:

- **id** - Job name

#### » **Import**

Glue Jobs can be imported using **name**, e.g.

```
$ terraform import aws_glue_job.MyJob MyJob
```

## » `aws__guardduty__detector`

Provides a resource to manage a GuardDuty detector.

**NOTE:** Deleting this resource is equivalent to "disabling" GuardDuty for an AWS region, which removes all existing findings. You can set the `enable` attribute to `false` to instead "suspend" monitoring and feedback reporting while keeping existing data. See the Suspending or Disabling Amazon GuardDuty documentation for more information.

### » Example Usage

```
resource "aws_guardduty_detector" "MyDetector" {  
  enable = true  
}
```

### » Argument Reference

The following arguments are supported:

- `enable` - (Optional) Enable monitoring and feedback reporting. Setting to `false` is equivalent to "suspending" GuardDuty. Defaults to `true`.

### » Attributes Reference

The following additional attributes are exported:

- `id` - The ID of the GuardDuty detector
- `account_id` - The AWS account ID of the GuardDuty detector

### » Import

GuardDuty detectors can be imported using the detector ID, e.g.

```
$ terraform import aws_guardduty_detector.MyDetector 00b00fd5aecc0ab60a708659477e9617
```

## » `aws__guardduty__ipset`

Provides a resource to manage a GuardDuty IPSet.

**Note:** Currently in GuardDuty, users from member accounts cannot upload and further manage IPSets. IPSets that are uploaded by the master account

are imposed on GuardDuty functionality in its member accounts. See the GuardDuty API Documentation

## » Example Usage

```
resource "aws_guardduty_detector" "master" {
  enable = true
}

resource "aws_s3_bucket" "bucket" {
  acl = "private"
}

resource "aws_s3_bucket_object" "MyIPSet" {
  acl      = "public-read"
  content  = "10.0.0.0/8\n"
  bucket   = "${aws_s3_bucket.bucket.id}"
  key      = "MyIPSet"
}

resource "aws_guardduty_ipset" "MyIPSet" {
  activate      = true
  detector_id   = "${aws_guardduty_detector.master.id}"
  format        = "TXT"
  location      = "https://s3.amazonaws.com/${aws_s3_bucket_object.MyIPSet.bucket}/${aws_s3_bu
  name          = "MyIPSet"
}
```

## » Argument Reference

The following arguments are supported:

- **activate** - (Required) Specifies whether GuardDuty is to start using the uploaded IPSet.
- **detector\_id** - (Required) The detector ID of the GuardDuty.
- **format** - (Required) The format of the file that contains the IPSet. Valid values: `TXT` | `STIX` | `OTX_CSV` | `ALIEN_VAULT` | `PROOF_POINT` | `FIRE_EYE`
- **location** - (Required) The URI of the file that contains the IPSet.
- **name** - (Required) The friendly name to identify the IPSet.

## » Attributes Reference

The following additional attributes are exported:

- `id` - The ID of the GuardDuty IPSet.

## » Import

GuardDuty IPSet can be imported using the the master GuardDuty detector ID and IPSet ID, e.g.

```
$ terraform import aws_guarddduty_ipset.MyIPSet 00b00fd5aecc0ab60a708659477e9617:123456789012
```

## » `aws__guarddduty__member`

Provides a resource to manage a GuardDuty member.

**NOTE:** Currently after using this resource, you must manually invite and accept member account invitations before GuardDuty will begin sending cross-account events. More information for how to accomplish this via the AWS Console or API can be found in the GuardDuty User Guide. Terraform implementation of member invitation and acceptance resources can be tracked in [Github](#).

## » Example Usage

```
resource "aws_guarddduty_detector" "master" {
  enable = true
}

resource "aws_guarddduty_detector" "member" {
  provider = "aws.dev"

  enable = true
}

resource "aws_guarddduty_member" "member" {
  account_id = "${aws_guarddduty_detector.member.account_id}"
  detector_id = "${aws_guarddduty_detector.master.id}"
  email      = "required@example.com"
}
```

## » Argument Reference

The following arguments are supported:

- `account_id` - (Required) AWS account ID for member account.

- `detector_id` - (Required) The detector ID of the GuardDuty account where you want to create member accounts.
- `email` - (Required) Email address for member account.

## » Attributes Reference

The following additional attributes are exported:

- `id` - The ID of the GuardDuty member

## » Import

GuardDuty members can be imported using the the master GuardDuty detector ID and member AWS account ID, e.g.

```
$ terraform import aws_guarddduty_member.MyMember 00b00fd5aecc0ab60a708659477e9617:1234567890
```

## » `aws_guarddduty__threatintelset`

Provides a resource to manage a GuardDuty ThreatIntelSet.

**Note:** Currently in GuardDuty, users from member accounts cannot upload and further manage ThreatIntelSets. ThreatIntelSets that are uploaded by the master account are imposed on GuardDuty functionality in its member accounts. See the GuardDuty API Documentation

## » Example Usage

```
resource "aws_guarddduty_detector" "master" {
  enable = true
}

resource "aws_s3_bucket" "bucket" {
  acl = "private"
}

resource "aws_s3_bucket_object" "MyThreatIntelSet" {
  acl      = "public-read"
  content  = "10.0.0.0/8\n"
  bucket   = "${aws_s3_bucket.bucket.id}"
  key      = "MyThreatIntelSet"
}
```

```
resource "aws_guardduty_threatintelset" "MyThreatIntelSet" {
  activate      = true
  detector_id   = "${aws_guardduty_detector.master.id}"
  format        = "TXT"
  location      = "https://s3.amazonaws.com/${aws_s3_bucket_object.MyThreatIntelSet.bucket}/${aws_guardduty_threatintelset.MyThreatIntelSet.name}"
  name          = "MyThreatIntelSet"
}
```

## » Argument Reference

The following arguments are supported:

- **activate** - (Required) Specifies whether GuardDuty is to start using the uploaded ThreatIntelSet.
- **detector\_id** - (Required) The detector ID of the GuardDuty.
- **format** - (Required) The format of the file that contains the ThreatIntelSet. Valid values: TXT | STIX | OTX\_CSV | ALIEN\_VAULT | PROOF\_POINT | FIRE\_EYE
- **location** - (Required) The URI of the file that contains the ThreatIntelSet.
- **name** - (Required) The friendly name to identify the ThreatIntelSet.

## » Attributes Reference

The following additional attributes are exported:

- **id** - The ID of the GuardDuty ThreatIntelSet and the detector ID. Format: <DetectorID>:<ThreatIntelSetID>

## » Import

GuardDuty ThreatIntelSet can be imported using the the master GuardDuty detector ID and ThreatIntelSetID, e.g.

```
$ terraform import aws_guardduty_threatintelset.MyThreatIntelSet 00b00fd5aecc0ab60a708659477
```

## » aws\_\_iam\_\_access\_\_key

Provides an IAM access key. This is a set of credentials that allow API requests to be made as an IAM user.

## » Example Usage

```
resource "aws_iam_access_key" "lb" {
  user      = "${aws_iam_user.lb.name}"
  pgp_key   = "keybase:some_person_that_exists"
}

resource "aws_iam_user" "lb" {
  name = "loadbalancer"
  path = "/system/"
}

resource "aws_iam_user_policy" "lb_ro" {
  name = "test"
  user = "${aws_iam_user.lb.name}"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:Describe*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
EOF
}

output "secret" {
  value = "${aws_iam_access_key.lb.encrypted_secret}"
}
```

## » Argument Reference

The following arguments are supported:

- **user** - (Required) The IAM user to associate with this access key.
- **pgp\_key** - (Optional) Either a base-64 encoded PGP public key, or a keybase username in the form `keybase:some_person_that_exists`.

## » Attributes Reference

The following attributes are exported:

- **id** - The access key ID.
- **user** - The IAM user associated with this access key.
- **key\_fingerprint** - The fingerprint of the PGP key used to encrypt the secret
- **secret** - The secret access key. Note that this will be written to the state file. Please supply a **pgp\_key** instead, which will prevent the secret from being stored in plain text
- **encrypted\_secret** - The encrypted secret, base64 encoded. ~> **NOTE:** The encrypted secret may be decrypted using the command line, for example: `terraform output secret | base64 --decode | keybase pgp decrypt`.
- **ses\_smtp\_password** - The secret access key converted into an SES SMTP password by applying AWS's documented conversion algorithm.
- **status** - "Active" or "Inactive". Keys are initially active, but can be made inactive by other means.

## » aws\_iam\_account\_alias

**Note:** There is only a single account alias per AWS account.

Manages the account alias for the AWS Account.

## » Example Usage

```
resource "aws_iam_account_alias" "alias" {  
  account_alias = "my-account-alias"  
}
```

## » Argument Reference

The following arguments are supported:

- **account\_alias** - (Required) The account alias

## » Import

The current Account Alias can be imported using the **account\_alias**, e.g.

```
$ terraform import aws_iam_account_alias.alias my-account-alias
```



## » `aws_iam_account_password_policy`

**Note:** There is only a single policy allowed per AWS account. An existing policy will be lost when using this resource as an effect of this limitation.

Manages Password Policy for the AWS Account. See more about Account Password Policy in the official AWS docs.

### » Example Usage

```
resource "aws_iam_account_password_policy" "strict" {
  minimum_password_length      = 8
  require_lowercase_characters = true
  require_numbers              = true
  require_uppercase_characters = true
  require_symbols              = true
  allow_users_to_change_password = true
}
```

### » Argument Reference

The following arguments are supported:

- `allow_users_to_change_password` - (Optional) Whether to allow users to change their own password
- `hard_expiry` - (Optional) Whether users are prevented from setting a new password after their password has expired (i.e. require administrator reset)
- `max_password_age` - (Optional) The number of days that an user password is valid.
- `minimum_password_length` - (Optional) Minimum length to require for user passwords.
- `password_reuse_prevention` - (Optional) The number of previous passwords that users are prevented from reusing.
- `require_lowercase_characters` - (Optional) Whether to require lowercase characters for user passwords.
- `require_numbers` - (Optional) Whether to require numbers for user passwords.
- `require_symbols` - (Optional) Whether to require symbols for user passwords.
- `require_uppercase_characters` - (Optional) Whether to require uppercase characters for user passwords.

## » Attributes Reference

The following attributes are exported:

- `expire_passwords` - Indicates whether passwords in the account expire. Returns `true` if `max_password_age` contains a value greater than 0. Returns `false` if it is 0 or *not present*.

## » Import

IAM Account Password Policy can be imported using the word `iam-account-password-policy`, e.g.

```
$ terraform import aws_iam_account_password_policy.strict iam-account-password-policy
```

## » `aws_iam_group`

Provides an IAM group.

## » Example Usage

```
resource "aws_iam_group" "developers" {  
  name = "developers"  
  path = "/users/"  
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) The group's name. The name must consist of upper and lowercase alphanumeric characters with no spaces. You can also include any of the following characters: `=, . @ - _ . .`. Group names are not distinguished by case. For example, you cannot create groups named both "ADMINS" and "admins".
- `path` - (Optional, default `"/"`) Path in which to create the group.

## » Attributes Reference

The following attributes are exported:

- `id` - The group's ID.

- **arn** - The ARN assigned by AWS for this group.
- **name** - The group's name.
- **path** - The path of the group in IAM.
- **unique\_id** - The unique ID assigned by AWS.

## » Import

IAM Groups can be imported using the **name**, e.g.

```
$ terraform import aws_iam_group.developers developers
```

## » aws\_iam\_group\_membership

Provides a top level resource to manage IAM Group membership for IAM Users. For more information on managing IAM Groups or IAM Users, see [IAM Groups](#) or [IAM Users](#)

**Note:** `aws_iam_group_membership` will conflict with itself if used more than once with the same group. To non-exclusively manage the users in a group, see the `aws_iam_user_group_membership` resource.

## » Example Usage

```
resource "aws_iam_group_membership" "team" {
  name = "tf-testing-group-membership"

  users = [
    "${aws_iam_user.user_one.name}",
    "${aws_iam_user.user_two.name}",
  ]

  group = "${aws_iam_group.group.name}"
}

resource "aws_iam_group" "group" {
  name = "test-group"
}

resource "aws_iam_user" "user_one" {
  name = "test-user"
}

resource "aws_iam_user" "user_two" {
```

```

    name = "test-user-two"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name to identify the Group Membership
- **users** - (Required) A list of IAM User names to associate with the Group
- **group** - (Required) The IAM Group name to attach the list of **users** to

## » Attributes Reference

- **name** - The name to identify the Group Membership
- **users** - list of IAM User names
- **group** - IAM Group name

## » aws\_iam\_group\_policy

Provides an IAM policy attached to a group.

## » Example Usage

```

resource "aws_iam_group_policy" "my_developer_policy" {
    name = "my_developer_policy"
    group = "${aws_iam_group.my_developers.id}"

    policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:Describe*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
EOF
}

```

```
resource "aws_iam_group" "my_developers" {
  name = "developers"
  path = "/users/"
}
```

## » Argument Reference

The following arguments are supported:

- **policy** - (Required) The policy document. This is a JSON formatted string. The heredoc syntax or `file` function is helpful here.
- **name** - (Optional) The name of the policy. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **group** - (Required) The IAM group to attach to the policy.

## » Attributes Reference

- **id** - The group policy ID.
- **group** - The group to which this policy applies.
- **name** - The name of the policy.
- **policy** - The policy document attached to the group.

## » aws\_iam\_group\_policy\_attachment

Attaches a Managed IAM Policy to an IAM group

## » Example Usage

```
resource "aws_iam_group" "group" {
  name = "test-group"
}

resource "aws_iam_policy" "policy" {
  name           = "test-policy"
  description    = "A test policy"
  policy        = # omitted
}

resource "aws_iam_group_policy_attachment" "test-attach" {
```

```

    group      = "${aws_iam_group.group.name}"
    policy_arn = "${aws_iam_policy.policy.arn}"
}

```

## » Argument Reference

The following arguments are supported:

- `group` (Required) - The group the policy should be applied to
- `policy_arn` (Required) - The ARN of the policy you want to apply

## » `aws_iam_instance_profile`

Provides an IAM instance profile.

**NOTE:** Either `role` or `roles` (**deprecated**) must be specified.

## » Example Usage

```

resource "aws_iam_instance_profile" "test_profile" {
  name = "test_profile"
  role = "${aws_iam_role.role.name}"
}

```

```

resource "aws_iam_role" "role" {
  name = "test_role"
  path = "/"
}

```

```

assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "ec2.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
EOF

```

}

## » Argument Reference

The following arguments are supported:

- **name** - (Optional, Forces new resource) The profile's name. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **path** - (Optional, default "/") Path in which to create the profile.
- **roles** - (**Deprecated**) A list of role names to include in the profile. The current default is 1. If you see an error message similar to **Cannot exceed quota for InstanceSessionsPerInstanceProfile: 1**, then you must contact AWS support and ask for a limit increase. **WARNING:** This is deprecated since version 0.9.3 (April 12, 2017), as  $\geq 2$  roles are not possible. See issue #11575.
- **role** - (Optional) The role name to include in the profile.

## » Attribute Reference

- **id** - The instance profile's ID.
- **arn** - The ARN assigned by AWS to the instance profile.
- **create\_date** - The creation timestamp of the instance profile.
- **name** - The instance profile's name.
- **path** - The path of the instance profile in IAM.
- **role** - The role assigned to the instance profile.
- **roles** - The list of roles assigned to the instance profile. (**Deprecated**)
- **unique\_id** - The unique ID assigned by AWS.

## » Import

Instance Profiles can be imported using the **name**, e.g.

```
$ terraform import aws_iam_instance_profile.test_profile app-instance-profile-1
```

## » `aws_iam_openid_connect_provider`

Provides an IAM OpenID Connect provider.

## » Example Usage

```
resource "aws_iam_openid_connect_provider" "default" {  
  url = "https://accounts.google.com"  
  client_id_list = [  
    "266362248691-342342xasdasdasda-apps.googleusercontent.com"  
  ]  
  thumbprint_list = []  
}
```

## » Argument Reference

The following arguments are supported:

- **url** - (Required) The URL of the identity provider. Corresponds to the *iss* claim.
- **client\_id\_list** - (Required) A list of client IDs (also known as audiences). When a mobile or web app registers with an OpenID Connect provider, they establish a value that identifies the application. (This is the value that's sent as the `client_id` parameter on OAuth requests.)
- **thumbprint\_list** - (Required) A list of server certificate thumbprints for the OpenID Connect (OIDC) identity provider's server certificate(s).

## » Attributes Reference

The following attributes are exported:

- **arn** - The ARN assigned by AWS for this provider.

## » Import

IAM OpenID Connect Providers can be imported using the **arn**, e.g.

```
$ terraform import aws_iam_openid_connect_provider.default arn:aws:iam::123456789012:oidc-pr
```

## » aws\_iam\_policy

Provides an IAM policy.



## » Example Usage

```
resource "aws_iam_policy" "policy" {
  name      = "test_policy"
  path      = "/"
  description = "My test policy"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:Describe*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
EOF
}
```

## » Argument Reference

The following arguments are supported:

- **description** - (Optional) Description of the IAM policy.
- **name** - (Optional, Forces new resource) The name of the policy. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **path** - (Optional, default `"/`) Path in which to create the policy. See IAM Identifiers for more information.
- **policy** - (Required) The policy document. This is a JSON formatted string. The heredoc syntax, `file` function, or the `aws_iam_policy_document` data source are all helpful here.

## » Attributes Reference

The following attributes are exported:

- **id** - The policy's ID.
- **arn** - The ARN assigned by AWS to this policy.
- **description** - The description of the policy.

- `name` - The name of the policy.
- `path` - The path of the policy in IAM.
- `policy` - The policy document.

## » Import

IAM Policies can be imported using the `arn`, e.g.

```
$ terraform import aws_iam_policy.administrator arn:aws:iam::123456789012:policy/UsersManager
```

## » `aws_iam_policy_attachment`

Attaches a Managed IAM Policy to user(s), role(s), and/or group(s)

**WARNING:** The `aws_iam_policy_attachment` resource creates **exclusive** attachments of IAM policies. Across the entire AWS account, all of the users/roles/groups to which a single policy is attached must be declared by a single `aws_iam_policy_attachment` resource. This means that even any users/roles/groups that have the attached policy via some mechanism other than Terraform will have that attached policy revoked by Terraform. Consider `aws_iam_role_policy_attachment`, `aws_iam_user_policy_attachment`, or `aws_iam_group_policy_attachment` instead. These resources do not enforce exclusive attachment of an IAM policy.

## » Example Usage

```
resource "aws_iam_user" "user" {
  name = "test-user"
}

resource "aws_iam_role" "role" {
  name = "test-role"
}

resource "aws_iam_group" "group" {
  name = "test-group"
}

resource "aws_iam_policy" "policy" {
  name           = "test-policy"
  description    = "A test policy"
  policy         = # omitted
}
```

```

resource "aws_iam_policy_attachment" "test-attach" {
  name      = "test-attachment"
  users     = ["${aws_iam_user.user.name}"]
  roles     = ["${aws_iam_role.role.name}"]
  groups    = ["${aws_iam_group.group.name}"]
  policy_arn = "${aws_iam_policy.policy.arn}"
}

```

## » Argument Reference

The following arguments are supported:

- **name** (Required) - The name of the attachment. This cannot be an empty string.
- **users** (Optional) - The user(s) the policy should be applied to
- **roles** (Optional) - The role(s) the policy should be applied to
- **groups** (Optional) - The group(s) the policy should be applied to
- **policy\_arn** (Required) - The ARN of the policy you want to apply

## » Attributes Reference

The following attributes are exported:

- **id** - The policy's ID.
- **name** - The name of the attachment.

## » aws\_iam\_role

Provides an IAM role.

## » Example Usage

```

resource "aws_iam_role" "test_role" {
  name = "test_role"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",

```

```

        "Principal": {
            "Service": "ec2.amazonaws.com"
        },
        "Effect": "Allow",
        "Sid": ""
    }
]
}
EOF
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional, Forces new resource) The name of the role. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **assume\_role\_policy** - (Required) The policy that grants an entity permission to assume the role.

**NOTE:** This **assume\_role\_policy** is very similar but slightly different than just a standard IAM policy and cannot use an **aws\_iam\_policy** resource. It *can* however, use an **aws\_iam\_policy\_document** data source, see example below for how this could work.

- **force\_detach\_policies** - (Optional) Specifies to force detaching any policies the role has before destroying it. Defaults to **false**.
- **path** - (Optional) The path to the role. See IAM Identifiers for more information.
- **description** - (Optional) The description of the role.
- **max\_session\_duration** - (Optional) The maximum session duration (in seconds) that you want to set for the specified role. If you do not specify a value for this setting, the default maximum of one hour is applied. This setting can have a value from 1 hour to 12 hours.

## » Attributes Reference

The following attributes are exported:

- **arn** - The Amazon Resource Name (ARN) specifying the role.
- **create\_date** - The creation date of the IAM role.
- **unique\_id** - The stable and unique string identifying the role.

- `name` - The name of the role.
- `description` - The description of the role.

## » Example of Using Data Source for Assume Role Policy

```
data "aws_iam_policy_document" "instance-assume-role-policy" {
  statement {
    actions = ["sts:AssumeRole"]

    principals {
      type       = "Service"
      identifiers = ["ec2.amazonaws.com"]
    }
  }
}

resource "aws_iam_role" "instance" {
  name       = "instance_role"
  path       = "/system/"
  assume_role_policy = "${data.aws_iam_policy_document.instance-assume-role-policy.json}"
}
```

## » Import

IAM Roles can be imported using the `name`, e.g.

```
$ terraform import aws_iam_role.developer developer_name
```

## » `aws_iam_role_policy`

Provides an IAM role policy.

## » Example Usage

```
resource "aws_iam_role_policy" "test_policy" {
  name = "test_policy"
  role = "${aws_iam_role.test_role.id}"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
```

```

    {
      "Action": [
        "ec2:Describe*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
EOF
}

resource "aws_iam_role" "test_role" {
  name = "test_role"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "ec2.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
}
EOF
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The name of the role policy. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **policy** - (Required) The policy document. This is a JSON formatted string. The heredoc syntax or `file` function is helpful here.
- **role** - (Required) The IAM role to attach to the policy.

## » Attributes Reference

- **id** - The role policy ID, in the form of `role_name:role_policy_name`.
- **name** - The name of the policy.
- **policy** - The policy document attached to the role.
- **role** - The name of the role associated with the policy.

## » Import

IAM Role Policies can be imported using the `role_name:role_policy_name`, e.g.

```
$ terraform import aws_iam_role_policy.mypolicy role_of_mypolicy_name:mypolicy_name
```

## » `aws_iam_role_policy_attachment`

Attaches a Managed IAM Policy to an IAM role

## » Example Usage

```
resource "aws_iam_role" "role" {
  name = "test-role"
  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "ec2.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
EOF
}

resource "aws_iam_policy" "policy" {
  name           = "test-policy"
  description    = "A test policy"
  policy = <<EOF
```

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:Describe*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
EOF
}

resource "aws_iam_role_policy_attachment" "test-attach" {
  role       = "${aws_iam_role.role.name}"
  policy_arn = "${aws_iam_policy.policy.arn}"
}

```

## » Argument Reference

The following arguments are supported:

- `role` (Required) - The role the policy should be applied to
- `policy_arn` (Required) - The ARN of the policy you want to apply

## » `aws_iam_saml_provider`

Provides an IAM SAML provider.

## » Example Usage

```

resource "aws_iam_saml_provider" "default" {
  name                = "myprovider"
  saml_metadata_document = "${file("saml-metadata.xml")}"
}

```

## » Argument Reference

The following arguments are supported:



- **name** - (Required) The name of the provider to create.
- **saml\_metadata\_document** - (Required) An XML document generated by an identity provider that supports SAML 2.0.

## » Attributes Reference

The following attributes are exported:

- **arn** - The ARN assigned by AWS for this provider.
- **valid\_until** - The expiration date and time for the SAML provider in RFC1123 format, e.g. `Mon, 02 Jan 2006 15:04:05 MST`.

## » Import

IAM SAML Providers can be imported using the **arn**, e.g.

```
$ terraform import aws_iam_saml_provider.default arn:aws:iam::123456789012:saml-provider/SAM
```

## » aws\_iam\_server\_certificate

Provides an IAM Server Certificate resource to upload Server Certificates. Certs uploaded to IAM can easily work with other AWS services such as:

- AWS Elastic Beanstalk
- Elastic Load Balancing
- CloudFront
- AWS OpsWorks

For information about server certificates in IAM, see Managing Server Certificates in AWS Documentation.

**Note:** All arguments including the private key will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Example Usage

Using certs on file:

```
resource "aws_iam_server_certificate" "test_cert" {
  name                = "some_test_cert"
  certificate_body     = "${file("self-ca-cert.pem")}"
  private_key         = "${file("test-key.pem")}"
}
```

Example with cert in-line:

```

resource "aws_iam_server_certificate" "test_cert_alt" {
  name = "alt_test_cert"

  certificate_body = <<EOF
-----BEGIN CERTIFICATE-----
[.....] # cert contents
-----END CERTIFICATE-----
EOF

  private_key = <<EOF
-----BEGIN RSA PRIVATE KEY-----
[.....] # cert contents
-----END RSA PRIVATE KEY-----
EOF
}

```

#### Use in combination with an AWS ELB resource:

Some properties of an IAM Server Certificates cannot be updated while they are in use. In order for Terraform to effectively manage a Certificate in this situation, it is recommended you utilize the `name_prefix` attribute and enable the `create_before_destroy` lifecycle block. This will allow Terraform to create a new, updated `aws_iam_server_certificate` resource and replace it in dependant resources before attempting to destroy the old version.

```

resource "aws_iam_server_certificate" "test_cert" {
  name_prefix      = "example-cert"
  certificate_body = "${file("self-ca-cert.pem")}"
  private_key      = "${file("test-key.pem")}"

  lifecycle {
    create_before_destroy = true
  }
}

resource "aws_elb" "ourapp" {
  name                = "terraform-asg-deployment-example"
  availability_zones  = ["us-west-2a"]
  cross_zone_load_balancing = true

  listener {
    instance_port      = 8000
    instance_protocol  = "http"
    lb_port            = 443
    lb_protocol        = "https"
    ssl_certificate_id = "${aws_iam_server_certificate.test_cert.arn}"
  }
}

```

}

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The name of the Server Certificate. Do not include the path in this value. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **certificate\_body** - (Required) The contents of the public key certificate in PEM-encoded format.
- **certificate\_chain** - (Optional) The contents of the certificate chain. This is typically a concatenation of the PEM-encoded public key certificates of the chain.
- **private\_key** - (Required) The contents of the private key in PEM-encoded format.
- **path** - (Optional) The IAM path for the server certificate. If it is not included, it defaults to a slash (/). If this certificate is for use with AWS CloudFront, the path must be in format `/cloudfront/your_path_here`. See IAM Identifiers for more details on IAM Paths.

**NOTE:** AWS performs behind-the-scenes modifications to some certificate files if they do not adhere to a specific format. These modifications will result in terraform forever believing that it needs to update the resources since the local and AWS file contents will not match after these modifications occur. In order to prevent this from happening you must ensure that all your PEM-encoded files use UNIX line-breaks and that **certificate\_body** contains only one certificate. All other certificates should go in **certificate\_chain**. It is common for some Certificate Authorities to issue certificate files that have DOS line-breaks and that are actually multiple certificates concatenated together in order to form a full certificate chain.

## » Attributes Reference

- **id** - The unique Server Certificate name
- **name** - The name of the Server Certificate
- **arn** - The Amazon Resource Name (ARN) specifying the server certificate.

## » Import

IAM Server Certificates can be imported using the **name**, e.g.

```
$ terraform import aws_iam_server_certificate.certificate example.com-certificate-until-2018
```

## » **aws\_iam\_service\_linked\_role**

Provides an IAM service-linked role.

### » **Example Usage**

```
resource "aws_iam_service_linked_role" "elasticbeanstalk" {  
  aws_service_name = "elasticbeanstalk.amazonaws.com"  
}
```

### » **Argument Reference**

The following arguments are supported:

- **aws\_service\_name** - (Required, Forces new resource) The AWS service to which this role is attached. You use a string similar to a URL but without the `http://` in front. For example: `elasticbeanstalk.amazonaws.com`. To find the full list of services that support service-linked roles, check the docs.
- **custom\_suffix** - (Optional, forces new resource) Additional string appended to the role name. Not all AWS services support custom suffixes.
- **description** - (Optional) The description of the role.

### » **Attributes Reference**

The following additional attributes are exported:

- **id** - The Amazon Resource Name (ARN) of the role.
- **arn** - The Amazon Resource Name (ARN) specifying the role.
- **create\_date** - The creation date of the IAM role.
- **name** - The name of the role.
- **path** - The path of the role.
- **unique\_id** - The stable and unique string identifying the role.

### » **Import**

IAM service-linked roles can be imported using role ARN, e.g.

```
$ terraform import aws_iam_service_linked_role.elasticbeanstalk arn:aws:iam::123456789012:ro
```

## » `aws_iam_user`

Provides an IAM user.

### » Example Usage

```
resource "aws_iam_user" "lb" {
  name = "loadbalancer"
  path = "/system/"
}

resource "aws_iam_access_key" "lb" {
  user = "${aws_iam_user.lb.name}"
}

resource "aws_iam_user_policy" "lb_ro" {
  name = "test"
  user = "${aws_iam_user.lb.name}"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:Describe*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
EOF
}
```

### » Argument Reference

The following arguments are supported:

- **name** - (Required) The user's name. The name must consist of upper and lowercase alphanumeric characters with no spaces. You can also include any of the following characters: `=, . @ - _ . .`. User names are not distinguished by case. For example, you cannot create users named both `"TESTUSER"` and `"testuser"`.

- **path** - (Optional, default `"/"`) Path in which to create the user.
- **force\_destroy** - (Optional, default `false`) When destroying this user, destroy even if it has non-Terraform-managed IAM access keys, login profile or MFA devices. Without **force\_destroy** a user with non-Terraform-managed access keys and login profile will fail to be destroyed.

## » Attributes Reference

The following attributes are exported:

- **arn** - The ARN assigned by AWS for this user.
- **name** - The user's name.
- **unique\_id** - The unique ID assigned by AWS.

## » Import

IAM Users can be imported using the **name**, e.g.

```
$ terraform import aws_iam_user.lb loadbalancer
```

## » `aws_iam_user_group_membership`

Provides a resource for adding an IAM User to IAM Groups. This resource can be used multiple times with the same user for non-overlapping groups.

To exclusively manage the users in a group, see the `aws_iam_group_membership` resource.

## » Example usage

```
resource "aws_iam_user_group_membership" {
  user = "${aws_iam_user.user1.name}"

  groups = [
    "${aws_iam_group.group1.name}",
    "${aws_iam_group.group2.name}",
  ]
}

resource "aws_iam_user_group_membership" {
  user = "${aws_iam_user.user1.name}"

  groups = [
```

```

        "${aws_iam_group.group3.name}",
    ]
}

resource "aws_iam_user" "user1" {
    name = "user1"
}

resource "aws_iam_group" "group1" {
    name = "group1"
}

resource "aws_iam_group" "group2" {
    name = "group2"
}

resource "aws_iam_group" "group3" {
    name = "group3"
}

```

## » Argument Reference

The following arguments are supported:

- **user** - (Required) The name of the IAM User to add to groups
- **groups** - (Required) A list of IAM Groups to add the user to

## » Attributes Reference

- **user** - The name of the IAM User
- **groups** - The list of IAM Groups

## » `aws_iam_user_login_profile`

Provides one-time creation of a IAM user login profile, and uses PGP to encrypt the password for safe transport to the user. PGP keys can be obtained from Keybase.

## » Example Usage

```

resource "aws_iam_user" "u" {
    name = "auser"
}

```

```

    path          = "/"
    force_destroy = true
  }

  resource "aws_iam_user_login_profile" "u" {
    user      = "${aws_iam_user.u.name}"
    pgp_key   = "keybase:some_person_that_exists"
  }

  output "password" {
    value = "${aws_iam_user_login_profile.u.encrypted_password}"
  }

```

## » Argument Reference

The following arguments are supported:

- **user** - (Required) The IAM user's name.
- **pgp\_key** - (Required) Either a base-64 encoded PGP public key, or a keybase username in the form `keybase:username`.
- **password\_reset\_required** - (Optional, default "true") Whether the user should be forced to reset the generated password on first login.
- **password\_length** - (Optional, default 20) The length of the generated password.

## » Attributes Reference

The following attributes are exported:

- **key\_fingerprint** - The fingerprint of the PGP key used to encrypt the password
- **encrypted\_password** - The encrypted password, base64 encoded.

**NOTE:** The encrypted password may be decrypted using the command line, for example: `terraform output password | base64 --decode | keybase pgp decrypt`.

## » Import

IAM Login Profiles may not be imported.



## » `aws_iam_user_policy`

Provides an IAM policy attached to a user.

### » Example Usage

```
resource "aws_iam_user_policy" "lb_ro" {
  name = "test"
  user = "${aws_iam_user.lb.name}"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:Describe*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
EOF
}

resource "aws_iam_user" "lb" {
  name = "loadbalancer"
  path = "/system/"
}

resource "aws_iam_access_key" "lb" {
  user = "${aws_iam_user.lb.name}"
}
```

### » Argument Reference

The following arguments are supported:

- `policy` - (Required) The policy document. This is a JSON formatted string. The heredoc syntax or `file` function is helpful here.
- `name` - (Optional) The name of the policy. If omitted, Terraform will assign a random, unique name.

- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **user** - (Required) IAM user to which to attach this policy.

## » Attributes Reference

- **id** - The user policy ID, in the form of **user\_name:user\_policy\_name**.
- **name** - The name of the policy (always set).

## » Import

IAM User Policies can be imported using the **user\_name:user\_policy\_name**, e.g.

```
$ terraform import aws_iam_user_policy.mypolicy user_of_mypolicy_name:mypolicy_name
```

## » aws\_iam\_user\_policy\_attachment

Attaches a Managed IAM Policy to an IAM user

## » Example Usage

```
resource "aws_iam_user" "user" {
  name = "test-user"
}

resource "aws_iam_policy" "policy" {
  name           = "test-policy"
  description    = "A test policy"
  policy        = # omitted
}

resource "aws_iam_user_policy_attachment" "test-attach" {
  user          = "${aws_iam_user.user.name}"
  policy_arn    = "${aws_iam_policy.policy.arn}"
}
```

## » Argument Reference

The following arguments are supported:

- **user** (Required) - The user the policy should be applied to

- `policy_arn` (Required) - The ARN of the policy you want to apply

## » `aws_iam_user_ssh_key`

Uploads an SSH public key and associates it with the specified IAM user.

### » Example Usage

```
resource "aws_iam_user" "user" {
  name = "test-user"
  path = "/"
}

resource "aws_iam_user_ssh_key" "user" {
  username    = "${aws_iam_user.user.name}"
  encoding    = "SSH"
  public_key  = "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQD3F6tyPEFEzVOLX3X8BsXdMsQz1x2cEikKDEY"
}
```

### » Argument Reference

The following arguments are supported:

- `username` - (Required) The name of the IAM user to associate the SSH public key with.
- `encoding` - (Required) Specifies the public key encoding format to use in the response. To retrieve the public key in ssh-rsa format, use `SSH`. To retrieve the public key in PEM format, use `PEM`.
- `public_key` - (Required) The SSH public key. The public key must be encoded in ssh-rsa format or PEM format.
- `status` - (Optional) The status to assign to the SSH public key. Active means the key can be used for authentication with an AWS CodeCommit repository. Inactive means the key cannot be used. Default is `active`.

### » Attributes Reference

The following attributes are exported:

- `ssh_public_key_id` - The unique identifier for the SSH public key.
- `fingerprint` - The MD5 message digest of the SSH public key.

## » **aws\_\_iot\_\_certificate**

Creates and manages an AWS IoT certificate.

### » **Example Usage**

```
resource "aws_iot_certificate" "cert" {
  csr = "${file("/my/csr.pem")}"
  active = true
}
```

### » **Argument Reference**

- **active** - (Required) Boolean flag to indicate if the certificate should be active
- **csr** - (Required) The certificate signing request. Review the IoT API Reference Guide for more information on creating a certificate from a certificate signing request (CSR).

### » **Attributes Reference**

- **arn** - The ARN of the created AWS IoT certificate

## » **aws\_\_iot\_\_policy**

Provides an IoT policy.

### » **Example Usage**

```
resource "aws_iot_policy" "pubsub" {
  name          = "PubSubToAnyTopic"
  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "iot:*"
      ],
      "Effect": "Allow",
```

```

        "Resource": "*"
    }
]
}
EOF
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the policy.
- **policy** - (Required) The policy document. This is a JSON formatted string. The heredoc syntax or **file** function is helpful here. Use the IoT Developer Guide for more information on IoT Policies

## » Attributes Reference

The following attributes are exported:

- **arn** - The ARN assigned by AWS to this policy.
- **name** - The name of this policy.
- **default\_version\_id** - The default version of this policy.
- **policy** - The policy document.

## » aws\_iam\_policy\_rule

## » Example Usage

```

resource "aws_iam_policy_rule" "rule" {
  name = "MyRule"
  description = "Example rule"
  enabled = true
  sql = "SELECT * FROM 'topic/test'"
  sql_version = "2015-10-08"

  sns {
    message_format = "RAW"
    role_arn = "${aws_iam_role.role.arn}"
    target_arn = "${aws_sns_topic.mytopic.arn}"
  }
}

```

```

resource "aws_sns_topic" "mytopic" {
  name = "mytopic"
}

resource "aws_iam_role" "role" {
  name = "myrole"
  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "iot.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
}
EOF
}

resource "aws_iam_role_policy" "iam_policy_for_lambda" {
  name = "mypolicy"
  role = "${aws_iam_role.role.id}"
  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sns:Publish"
      ],
      "Resource": "${aws_sns_topic.mytopic.arn}"
    }
  ]
}
}
EOF
}

```

## » Argument Reference

- **name** - (Required) The name of the rule.
- **description** - (Optional) The description of the rule.

- **enabled** - (Required) Specifies whether the rule is enabled.
- **sql** - (Required) The SQL statement used to query the topic. For more information, see AWS IoT SQL Reference (<http://docs.aws.amazon.com/iot/latest/developerguide/iot-rules.html#aws-iot-sql-reference>) in the AWS IoT Developer Guide.
- **sql\_version** - (Required) The version of the SQL rules engine to use when evaluating the rule.

The **cloudwatch\_alarm** object takes the following arguments:

- **alarm\_name** - (Required) The CloudWatch alarm name.
- **role\_arn** - (Required) The IAM role ARN that allows access to the CloudWatch alarm.
- **state\_reason** - (Required) The reason for the alarm change.
- **state\_value** - (Required) The value of the alarm state. Acceptable values are: OK, ALARM, INSUFFICIENT\_DATA.

The **cloudwatch\_metric** object takes the following arguments:

- **metric\_name** - (Required) The CloudWatch metric name.
- **metric\_namespace** - (Required) The CloudWatch metric namespace name.
- **metric\_timestamp** - (Optional) An optional Unix timestamp ([http://docs.aws.amazon.com/AmazonCloudWatch/latest/DeveloperGuide/cloudwatch\\_concepts.html#about\\_timestamp](http://docs.aws.amazon.com/AmazonCloudWatch/latest/DeveloperGuide/cloudwatch_concepts.html#about_timestamp)).
- **metric\_unit** - (Required) The metric unit (supported units can be found here: [http://docs.aws.amazon.com/AmazonCloudWatch/latest/DeveloperGuide/cloudwatch\\_concepts.html#Unit](http://docs.aws.amazon.com/AmazonCloudWatch/latest/DeveloperGuide/cloudwatch_concepts.html#Unit))
- **metric\_value** - (Required) The CloudWatch metric value.
- **role\_arn** - (Required) The IAM role ARN that allows access to the CloudWatch metric.

The **dynamodb** object takes the following arguments:

- **hash\_key\_field** - (Required) The hash key name.
- **hash\_key\_type** - (Optional) The hash key type. Valid values are "STRING" or "NUMBER".
- **hash\_key\_value** - (Required) The hash key value.
- **payload\_field** - (Optional) The action payload.
- **range\_key\_field** - (Optional) The range key name.
- **range\_key\_type** - (Optional) The range key type. Valid values are "STRING" or "NUMBER".
- **range\_key\_value** - (Optional) The range key value.
- **role\_arn** - (Required) The ARN of the IAM role that grants access to the DynamoDB table.
- **table\_name** - (Required) The name of the DynamoDB table.

The **elasticsearch** object takes the following arguments:

- **endpoint** - (Required) The endpoint of your Elasticsearch domain.

- **id** - (Required) The unique identifier for the document you are storing.
- **index** - (Required) The Elasticsearch index where you want to store your data.
- **role\_arn** - (Required) The IAM role ARN that has access to Elasticsearch.
- **type** - (Required) The type of document you are storing.

The **firehose** object takes the following arguments:

- **delivery\_stream\_name** - (Required) The delivery stream name.
- **role\_arn** - (Required) The IAM role ARN that grants access to the Amazon Kinesis Firehose stream.

The **kinesis** object takes the following arguments:

- **partition\_key** - (Optional) The partition key.
- **role\_arn** - (Required) The ARN of the IAM role that grants access to the Amazon Kinesis stream.
- **stream\_name** - (Required) The name of the Amazon Kinesis stream.

The **lambda** object takes the following arguments:

- **function\_arn** - (Required) The ARN of the Lambda function.

The **republish** object takes the following arguments:

- **role\_arn** - (Required) The ARN of the IAM role that grants access.
- **topic** - (Required) The name of the MQTT topic the message should be republished to.

The **s3** object takes the following arguments:

- **bucket\_name** - (Required) The Amazon S3 bucket name.
- **key** - (Required) The object key.
- **role\_arn** - (Required) The ARN of the IAM role that grants access.

The **sns** object takes the following arguments:

- **message\_format** - (Required) The message format of the message to publish. Accepted values are "JSON" and "RAW".
- **role\_arn** - (Required) The ARN of the IAM role that grants access.
- **target\_arn** - (Required) The ARN of the SNS topic.

The **sqs** object takes the following arguments:

- **queue\_url** - (Required) The URL of the Amazon SQS queue.
- **role\_arn** - (Required) The ARN of the IAM role that grants access.
- **use\_base64** - (Required) Specifies whether to use Base64 encoding.

## » Attributes Reference

In addition to all arguments above, the following attributes are exported:



- `id` - The name of the topic rule
- `arn` - The ARN of the topic rule

## » Import

IoT Topic Rules can be imported using the `name`, e.g.

```
$ terraform import aws_iot_topic_rule.rule <name>
```

## » `aws__iot__thing`

Creates and manages an AWS IoT Thing.

## » Example Usage

```
resource "aws_iot_thing" "example" {
  name = "example"
  attributes {
    First = "examplevalue"
  }
}
```

## » Argument Reference

- `name` - (Required) The name of the thing.
- `attributes` - (Optional) Map of attributes of the thing.
- `thing_type_name` - (Optional) The thing type name.

## » Attributes Reference

In addition to the arguments above, the following attributes are exported:

- `default_client_id` - The default client ID.
- `version` - The current version of the thing record in the registry.
- `arn` - The ARN of the thing.

## » Import

IOT Things can be imported using the name, e.g.

```
$ terraform import aws_iot_thing.example example
```

## » **aws\_\_iot\_\_thing\_\_type**

Creates and manages an AWS IoT Thing Type.

### » **Example Usage**

```
resource "aws_iot_thing_type" "foo" {  
  name = "my_iot_thing"  
}
```

### » **Argument Reference**

- **name** - (Required, Forces New Resource) The name of the thing type.
- **description** - (Optional, Forces New Resource) The description of the thing type.
- **deprecated** - (Optional, Defaults to false) Whether the thing type is deprecated. If true, no new things could be associated with this type.
- **searchable\_attributes** - (Optional, Forces New Resource) A list of searchable thing attribute names.

### » **Attributes Reference**

In addition to the arguments above, the following attributes are exported:

- **arn** - The ARN of the created AWS IoT Thing Type.

## » **aws\_\_inspector\_\_assessment\_\_target**

Provides a Inspector assessment target

### » **Example Usage**

```
resource "aws_inspector_resource_group" "bar" {  
  tags {  
    Name = "foo"  
    Env  = "bar"  
  }  
}  
  
resource "aws_inspector_assessment_target" "foo" {  
  name = "assessment target"}
```

```

    resource_group_arn = "${aws_inspector_resource_group.bar.arn}"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the assessment target.
- **resource\_group\_arn** (Required )- The resource group ARN stating tags for instance matching.

## » Attributes Reference

The following attributes are exported:

- **arn** - The target assessment ARN.

## » aws\_inspector\_assessment\_template

Provides a Inspector assessment template

## » Example Usage

```

resource "aws_inspector_assessment_template" "foo" {
  name          = "bar template"
  target_arn    = "${aws_inspector_assessment_target.foo.arn}"
  duration      = 3600

  rules_package_arns = [
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p",
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-H5hpSawc",
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-JJ0tZiqQ",
    "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-vg5GGHSD",
  ]
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the assessment template.

- **target\_arn** - (Required) The assessment target ARN to attach the template to.
- **duration** - (Required) The duration of the inspector run.
- **rules\_package\_arns** - (Required) The rules to be used during the run.

## » Attributes Reference

The following attributes are exported:

- **arn** - The template assessment ARN.

## » **aws\_inspector\_resource\_group**

Provides a Inspector resource group

## » Example Usage

```
resource "aws_inspector_resource_group" "bar" {
  tags {
    Name = "foo"
    Env  = "bar"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **tags** - (Required) The tags on your EC2 Instance.

## » Attributes Reference

The following attributes are exported:

- **arn** - The resource group ARN.

## » **aws\_kinesis\_stream**

Provides a Kinesis Stream resource. Amazon Kinesis is a managed service that scales elastically for real-time processing of streaming big data.

For more details, see the Amazon Kinesis Documentation.

## » Example Usage

```
resource "aws_kinesis_stream" "test_stream" {
  name           = "terraform-kinesis-test"
  shard_count    = 1
  retention_period = 48

  shard_level_metrics = [
    "IncomingBytes",
    "OutgoingBytes",
  ]

  tags {
    Environment = "test"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A name to identify the stream. This is unique to the AWS account and region the Stream is created in.
- **shard\_count** - (Required) The number of shards that the stream will use. Amazon has guidelines for specifying the Stream size that should be referenced when creating a Kinesis stream. See Amazon Kinesis Streams for more.
- **retention\_period** - (Optional) Length of time data records are accessible after they are added to the stream. The maximum value of a stream's retention period is 168 hours. Minimum value is 24. Default is 24.
- **shard\_level\_metrics** - (Optional) A list of shard-level CloudWatch metrics which can be enabled for the stream. See Monitoring with CloudWatch for more. Note that the value ALL should not be used; instead you should provide an explicit list of metrics you wish to enable.
- **encryption\_type** - (Optional) The encryption type to use. The only acceptable values are NONE or KMS. The default value is NONE.
- **kms\_key\_id** - (Optional) The GUID for the customer-managed KMS key to use for encryption. You can also use a Kinesis-owned master key by specifying the alias aws/kinesis.
- **tags** - (Optional) A mapping of tags to assign to the resource.

## » Attributes Reference

- `id` - The unique Stream id
- `name` - The unique Stream name
- `shard_count` - The count of Shards for this Stream
- `arn` - The Amazon Resource Name (ARN) specifying the Stream (same as `id`)

## » Timeouts

`aws_kinesis_stream` provides the following Timeouts configuration options:

- `create` - (Default 5 minutes) Used for Creating a Kinesis Stream
- `update` - (Default 120 minutes) Used for Updating a Kinesis Stream
- `delete` - (Default 120 minutes) Used for Destroying a Kinesis Stream

## » Import

Kinesis Streams can be imported using the `name`, e.g.

```
$ terraform import aws_kinesis_stream.test_stream terraform-kinesis-test
```

## » `aws_kinesis_firehose_delivery_stream`

Provides a Kinesis Firehose Delivery Stream resource. Amazon Kinesis Firehose is a fully managed, elastic service to easily deliver real-time data streams to destinations such as Amazon S3 and Amazon Redshift.

For more details, see the Amazon Kinesis Firehose Documentation.

## » Example Usage

### » Extended S3 Destination

```
resource "aws_kinesis_firehose_delivery_stream" "extended_s3_stream" {
  name          = "terraform-kinesis-firehose-extended-s3-test-stream"
  destination   = "extended_s3"

  extended_s3_configuration {
    role_arn    = "${aws_iam_role.firehose_role.arn}"
    bucket_arn = "${aws_s3_bucket.bucket.arn}"
    processing_configuration = [
      {

```

```

        enabled = "true"
        processors = [
            {
                type = "Lambda"
                parameters = [
                    {
                        parameter_name = "LambdaArn"
                        parameter_value = "${aws_lambda_function.lambda_processor.arn}:$LATEST"
                    }
                ]
            }
        ]
    }
}

resource "aws_s3_bucket" "bucket" {
    bucket = "tf-test-bucket"
    acl    = "private"
}

resource "aws_iam_role" "firehose_role" {
    name = "firehose_test_role"

    assume_role_policy = <<EOF
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "sts:AssumeRole",
            "Principal": {
                "Service": "firehose.amazonaws.com"
            },
            "Effect": "Allow",
            "Sid": ""
        }
    ]
}
EOF
}

resource "aws_iam_role" "lambda_iam" {
    name = "lambda_iam"
    assume_role_policy = <<EOF
{

```

```

    "Version": "2012-10-17",
    "Statement": [
      {
        "Action": "sts:AssumeRole",
        "Principal": {
          "Service": "lambda.amazonaws.com"
        },
        "Effect": "Allow",
        "Sid": ""
      }
    ]
  }
}
EOF
}

resource "aws_lambda_function" "lambda_processor" {
  filename = "lambda.zip"
  function_name = "firehose_lambda_processor"
  role = "${aws_iam_role.lambda_iam.arn}"
  handler = "exports.handler"
  runtime = "nodejs4.3"
}

```

## » S3 Destination

```

resource "aws_s3_bucket" "bucket" {
  bucket = "tf-test-bucket"
  acl    = "private"
}

resource "aws_iam_role" "firehose_role" {
  name = "firehose_test_role"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "firehose.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
EOF
}

```



```

    ]
  }
EOF
}

resource "aws_kinesis_firehose_delivery_stream" "test_stream" {
  name          = "terraform-kinesis-firehose-test-stream"
  destination    = "s3"

  s3_configuration {
    role_arn      = "${aws_iam_role.firehose_role.arn}"
    bucket_arn    = "${aws_s3_bucket.bucket.arn}"
  }
}

```

## » Redshift Destination

```

resource "aws_redshift_cluster" "test_cluster" {
  cluster_identifier = "tf-redshift-cluster-%d"
  database_name      = "test"
  master_username    = "testuser"
  master_password    = "T3stPass"
  node_type          = "dc1.large"
  cluster_type       = "single-node"
}

resource "aws_kinesis_firehose_delivery_stream" "test_stream" {
  name          = "terraform-kinesis-firehose-test-stream"
  destination    = "redshift"

  s3_configuration {
    role_arn          = "${aws_iam_role.firehose_role.arn}"
    bucket_arn        = "${aws_s3_bucket.bucket.arn}"
    buffer_size       = 10
    buffer_interval    = 400
    compression_format = "GZIP"
  }

  redshift_configuration {
    role_arn          = "${aws_iam_role.firehose_role.arn}"
    cluster_jdbcurl    = "jdbc:redshift://${aws_redshift_cluster.test_cluster.endpoint}/${aws_iam_role.firehose_role.arn}"
    username          = "testuser"
    password          = "T3stPass"
    data_table_name    = "test-table"
    copy_options       = "delimiter '|' # the default delimiter"
  }
}

```

```

    data_table_columns = "test-col"
    s3_backup_mode      = "Enabled"
    s3_backup_configuration {
        role_arn          = "${aws_iam_role.firehose_role.arn}"
        bucket_arn        = "${aws_s3_bucket.bucket.arn}"
        buffer_size       = 15
        buffer_interval    = 300
        compression_format = "GZIP"
    }
}
}

```

## » Elasticsearch Destination

```

resource "aws_elasticsearch_domain" "test_cluster" {
    domain_name = "firehose-es-test"
}

resource "aws_kinesis_firehose_delivery_stream" "test_stream" {
    name          = "terraform-kinesis-firehose-test-stream"
    destination   = "elasticsearch"

    s3_configuration {
        role_arn          = "${aws_iam_role.firehose_role.arn}"
        bucket_arn        = "${aws_s3_bucket.bucket.arn}"
        buffer_size       = 10
        buffer_interval    = 400
        compression_format = "GZIP"
    }

    elasticsearch_configuration {
        domain_arn = "${aws_elasticsearch_domain.test_cluster.arn}"
        role_arn   = "${aws_iam_role.firehose_role.arn}"
        index_name = "test"
        type_name  = "test"

        processing_configuration = [
            {
                enabled = "true"
                processors = [
                    {
                        type = "Lambda"
                        parameters = [
                            {
                                parameter_name = "LambdaArn"

```



- **destination** – (Required) This is the destination to where the data is delivered. The only options are **s3** (Deprecated, use **extended\_s3** instead), **extended\_s3**, **redshift**, and **elasticsearch**.
- **s3\_configuration** - (Optional, Deprecated, see/use **extended\_s3\_configuration** unless **destination** is **redshift**) Configuration options for the s3 destination (or the intermediate bucket if the destination is redshift). More details are given below.
- **extended\_s3\_configuration** - (Optional, only Required when **destination** is **extended\_s3**) Enhanced configuration options for the s3 destination. More details are given below.
- **redshift\_configuration** - (Optional) Configuration options if redshift is the destination. Using **redshift\_configuration** requires the user to also specify a **s3\_configuration** block. More details are given below.

The **kinesis\_source\_configuration** object supports the following: \* **kinesis\_stream\_arn** (Required) The kinesis stream used as the source of the firehose delivery stream. \* **role\_arn** (Required) The ARN of the role that provides access to the source Kinesis stream.

The **s3\_configuration** object supports the following:

- **role\_arn** - (Required) The ARN of the AWS credentials.
- **bucket\_arn** - (Required) The ARN of the S3 bucket
- **prefix** - (Optional) The "YYYY/MM/DD/HH" time format prefix is automatically used for delivered S3 files. You can specify an extra prefix to be added in front of the time format prefix. Note that if the prefix ends with a slash, it appears as a folder in the S3 bucket
- **buffer\_size** - (Optional) Buffer incoming data to the specified size, in MBs, before delivering it to the destination. The default value is 5. We recommend setting **SizeInMBs** to a value greater than the amount of data you typically ingest into the delivery stream in 10 seconds. For example, if you typically ingest data at 1 MB/sec set **SizeInMBs** to be 10 MB or higher.
- **buffer\_interval** - (Optional) Buffer incoming data for the specified period of time, in seconds, before delivering it to the destination. The default value is 300.
- **compression\_format** - (Optional) The compression format. If no value is specified, the default is **UNCOMPRESSED**. Other supported values are **GZIP**, **ZIP** & **Snappy**. If the destination is redshift you cannot use **ZIP** or **Snappy**.
- **kms\_key\_arn** - (Optional) Specifies the KMS key ARN the stream will use to encrypt data. If not set, no encryption will be used.
- **cloudwatch\_logging\_options** - (Optional) The CloudWatch Logging Options for the delivery stream. More details are given below

The **extended\_s3\_configuration** object supports the same fields from **s3\_configuration** as well as the following:

- **processing\_configuration** - (Optional) The data processing configuration. More details are given below.
- **s3\_backup\_mode** - (Optional) The Amazon S3 backup mode. Valid values are **Disabled** and **Enabled**. Default value is **Disabled**.
- **s3\_backup\_configuration** - (Optional) The configuration for backup in Amazon S3. Required if **s3\_backup\_mode** is **Enabled**. Supports the same fields as **s3\_configuration** object.

The **redshift\_configuration** object supports the following:

- **cluster\_jdbcurl** - (Required) The jdbcurl of the redshift cluster.
- **username** - (Required) The username that the firehose delivery stream will assume. It is strongly recommended that the username and password provided is used exclusively for Amazon Kinesis Firehose purposes, and that the permissions for the account are restricted for Amazon Redshift INSERT permissions.
- **password** - (Required) The password for the username above.
- **retry\_duration** - (Optional) The length of time during which Firehose retries delivery after a failure, starting from the initial request and including the first attempt. The default value is 3600 seconds (60 minutes). Firehose does not retry if the value of **DurationInSeconds** is 0 (zero) or if the first delivery attempt takes longer than the current value.
- **role\_arn** - (Required) The arn of the role the stream assumes.
- **s3\_backup\_mode** - (Optional) The Amazon S3 backup mode. Valid values are **Disabled** and **Enabled**. Default value is **Disabled**.
- **s3\_backup\_configuration** - (Optional) The configuration for backup in Amazon S3. Required if **s3\_backup\_mode** is **Enabled**. Supports the same fields as **s3\_configuration** object.
- **data\_table\_name** - (Required) The name of the table in the redshift cluster that the s3 bucket will copy to.
- **copy\_options** - (Optional) Copy options for copying the data from the s3 intermediate bucket into redshift, for example to change the default delimiter. For valid values, see the AWS documentation
- **data\_table\_columns** - (Optional) The data table columns that will be targeted by the copy command.
- **cloudwatch\_logging\_options** - (Optional) The CloudWatch Logging Options for the delivery stream. More details are given below
- **processing\_configuration** - (Optional) The data processing configuration. More details are given below.

The **elasticsearch\_configuration** object supports the following:

- **buffering\_interval** - (Optional) Buffer incoming data for the specified period of time, in seconds between 60 to 900, before delivering it to the destination. The default value is 300s.
- **buffering\_size** - (Optional) Buffer incoming data to the specified size, in MBs between 1 to 100, before delivering it to the destination. The default value is 5MB.

- **domain\_arn** - (Required) The ARN of the Amazon ES domain. The IAM role must have permission for `DescribeElasticsearchDomain`, `DescribeElasticsearchDomains`, and `DescribeElasticsearchDomainConfig` after assuming `RoleARN`. The pattern needs to be `arn:.*`.
- **index\_name** - (Required) The Elasticsearch index name.
- **index\_rotation\_period** - (Optional) The Elasticsearch index rotation period. Index rotation appends a timestamp to the `IndexName` to facilitate expiration of old data. Valid values are `NoRotation`, `OneHour`, `OneDay`, `OneWeek`, and `OneMonth`. The default value is `OneDay`.
- **retry\_duration** - (Optional) After an initial failure to deliver to Amazon Elasticsearch, the total amount of time, in seconds between 0 to 7200, during which Firehose re-attempts delivery (including the first attempt). After this time has elapsed, the failed documents are written to Amazon S3. The default value is 300s. There will be no retry if the value is 0.
- **role\_arn** - (Required) The ARN of the IAM role to be assumed by Firehose for calling the Amazon ES Configuration API and for indexing documents. The pattern needs to be `arn:.*`.
- **s3\_backup\_mode** - (Optional) Defines how documents should be delivered to Amazon S3. Valid values are `FailedDocumentsOnly` and `AllDocuments`. Default value is `FailedDocumentsOnly`.
- **type\_name** - (Required) The Elasticsearch type name with maximum length of 100 characters.
- **cloudwatch\_logging\_options** - (Optional) The CloudWatch Logging Options for the delivery stream. More details are given below
- **processing\_configuration** - (Optional) The data processing configuration. More details are given below.

The `splunk_configuration` object supports the following:

- **hec\_acknowledgment\_timeout** - (Optional) The amount of time, in seconds between 180 and 600, that Kinesis Firehose waits to receive an acknowledgment from Splunk after it sends it data.
- **hec\_endpoint** - (Required) The HTTP Event Collector (HEC) endpoint to which Kinesis Firehose sends your data.
- **hec\_endpoint\_type** - (Optional) The HEC endpoint type. Valid values are `Raw` or `Event`. The default value is `Raw`.
- **hec\_token** - The GUID that you obtain from your Splunk cluster when you create a new HEC endpoint.
- **s3\_backup\_mode** - (Optional) Defines how documents should be delivered to Amazon S3. Valid values are `FailedEventsOnly` and `AllEvents`. Default value is `FailedEventsOnly`.
- **retry\_duration** - (Optional) After an initial failure to deliver to Amazon Elasticsearch, the total amount of time, in seconds between 0 to 7200, during which Firehose re-attempts delivery (including the first attempt). After this time has elapsed, the failed documents are written to Amazon S3. The default value is 300s. There will be no retry if the value is 0.
- **cloudwatch\_logging\_options** - (Optional) The CloudWatch Logging

Options for the delivery stream. More details are given below.

The `cloudwatch_logging_options` object supports the following:

- **enabled** - (Optional) Enables or disables the logging. Defaults to **false**.
- **log\_group\_name** - (Optional) The CloudWatch group name for logging. This value is required if **enabled** is true.
- **log\_stream\_name** - (Optional) The CloudWatch log stream name for logging. This value is required if **enabled** is true.

The `processing_configuration` object supports the following:

- **enabled** - (Optional) Enables or disables data processing.
- **processors** - (Optional) Array of data processors. More details are given below

The `processors` array objects support the following:

- **type** - (Required) The type of processor. Valid Values: **Lambda**
- **parameters** - (Optional) Array of processor parameters. More details are given below

The `parameters` array objects support the following:

- **parameter\_name** - (Required) Parameter name. Valid Values: **LambdaArn**, **NumberOfRetries**, **RoleArn**, **BufferSizeInMBs**, **BufferIntervalInSeconds**
- **parameter\_value** - (Required) Parameter value. Must be between 1 and 512 length (inclusive). When providing a Lambda ARN, you should specify the resource version as well.

## » Attributes Reference

- **arn** - The Amazon Resource Name (ARN) specifying the Stream

## » Import

Kinesis Firehose Delivery streams can be imported using the stream ARN, e.g.

```
$ terraform import aws_kinesis_firehose_delivery_stream.foo arn:aws:firehose:us-east-1:XXX:delivery-stream:foo
```

Note: Import does not work for stream destination **s3**. Consider using **extended\_s3** since **s3** destination is deprecated.

## » `aws_kms_alias`

Provides an alias for a KMS customer master key. AWS Console enforces 1-to-1 mapping between aliases & keys, but API (hence Terraform too) allows you to

create as many aliases as the account limits allow you.

## » Example Usage

```
resource "aws_kms_key" "a" {}

resource "aws_kms_alias" "a" {
  name           = "alias/my-key-alias"
  target_key_id = "${aws_kms_key.a.key_id}"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The display name of the alias. The name must start with the word "alias" followed by a forward slash (alias/)
- **name\_prefix** - (Optional) Creates an unique alias beginning with the specified prefix. The name must start with the word "alias" followed by a forward slash (alias/). Conflicts with **name**.
- **target\_key\_id** - (Required) Identifier for the key for which the alias is for, can be either an ARN or key\_id.

## » Attributes Reference

In addition to the arguments, the following attributes are exported:

- **arn** - The Amazon Resource Name (ARN) of the key alias.
- **target\_key\_arn** - The Amazon Resource Name (ARN) of the target key identifier.

## » Import

KMS aliases can be imported using the **name**, e.g.

```
$ terraform import aws_kms_alias.a alias/my-key-alias
```

## » aws\_kms\_grant

Provides a resource-based access control mechanism for a KMS customer master key.



## » Example Usage

```
resource "aws_kms_key" "a" {}

resource "aws_iam_role" "a" {
  name = "iam-role-for-grant"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "lambda.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
EOF
}

resource "aws_kms_grant" "a" {
  name           = "my-grant"
  key_id         = "${aws_kms_key.a.key_id}"
  grantee_principal = "${aws_iam_role.a.arn}"
  operations      = [ "Encrypt", "Decrypt", "GenerateDataKey" ]
  constraints {
    encryption_context_equals {
      Department = "Finance"
    }
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional, Forces new resources) A friendly name for identifying the grant.
- **key\_id** - (Required, Forces new resources) The unique identifier for the customer master key (CMK) that the grant applies to.
- **grantee\_principal** - (Required, Forces new resources) The principal that

is given permission to perform the operations that the grant permits in ARN format. Note that due to eventual consistency issues around IAM principals, terraform's state may not always be refreshed to reflect what is true in AWS.

- **operations** - (Required, Forces new resources) A list of operations that the grant permits. The permitted values are: `Decrypt`, `Encrypt`, `GenerateDataKey`, `GenerateDataKeyWithoutPlaintext`, `ReEncryptFrom`, `ReEncryptTo`, `CreateGrant`, `RetireGrant`, `DescribeKey`
- **retiree\_principal** - (Optional, Forces new resources) The principal that is given permission to retire the grant by using `RetireGrant` operation in ARN format. Note that due to eventual consistency issues around IAM principals, terraform's state may not always be refreshed to reflect what is true in AWS.
- **constraints** - (Optional, Forces new resources) A structure that you can use to allow certain operations in the grant only when the desired encryption context is present. For more information about encryption context, see Encryption Context.
- **grant\_creation\_tokens** - (Optional, Forces new resources) A list of grant tokens to be used when creating the grant. See Grant Tokens for more information about grant tokens.
- **retire\_on\_delete** - (Defaults to false, Forces new resources) If set to false (the default) the grants will be revoked upon deletion, and if set to true the grants will try to be retired upon deletion. Note that retiring grants requires special permissions, hence why we default to revoking grants. See `RetireGrant` for more information.

The **constraints** block supports the following arguments:

- **encryption\_context\_equals** - (Optional) A list of key-value pairs that must be present in the encryption context of certain subsequent operations that the grant allows. Conflicts with **encryption\_context\_subset**.
- **encryption\_context\_subset** - (Optional) A list of key-value pairs, all of which must be present in the encryption context of certain subsequent operations that the grant allows. Conflicts with **encryption\_context\_equals**.

## » Attributes Reference

The following attributes are exported:

- **grant\_id** - The unique identifier for the grant.
- **grant\_token** - The grant token for the created grant. For more information, see Grant Tokens.

## » **aws\_kms\_key**

Provides a KMS customer master key.

### » **Example Usage**

```
resource "aws_kms_key" "a" {  
  description          = "KMS key 1"  
  deletion_window_in_days = 10  
}
```

### » **Argument Reference**

The following arguments are supported:

- **description** - (Optional) The description of the key as viewed in AWS console.
- **key\_usage** - (Optional) Specifies the intended use of the key. Defaults to ENCRYPT\_DECRYPT, and only symmetric encryption and decryption are supported.
- **policy** - (Optional) A valid policy JSON document.
- **deletion\_window\_in\_days** - (Optional) Duration in days after which the key is deleted after destruction of the resource, must be between 7 and 30 days. Defaults to 30 days.
- **is\_enabled** - (Optional) Specifies whether the key is enabled. Defaults to true.
- **enable\_key\_rotation** - (Optional) Specifies whether key rotation is enabled. Defaults to false.
- **tags** - (Optional) A mapping of tags to assign to the object.

### » **Attributes Reference**

The following attributes are exported:

- **arn** - The Amazon Resource Name (ARN) of the key.
- **key\_id** - The globally unique identifier for the key.

### » **Import**

KMS Keys can be imported using the id, e.g.

```
$ terraform import aws_kms_key.a arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-5
```

## » **aws\_lambda\_alias**

Creates a Lambda function alias. Creates an alias that points to the specified Lambda function version.

For information about Lambda and how to use it, see [What is AWS Lambda?](#)  
For information about function aliases, see [CreateAlias](#) in the API docs.

### » **Example Usage**

```
resource "aws_lambda_alias" "test_alias" {  
  name           = "testalias"  
  description    = "a sample description"  
  function_name  = "${aws_lambda_function.lambda_function_test.arn}"  
  function_version = "$LATEST"  
}
```

### » **Argument Reference**

- **name** - (Required) Name for the alias you are creating. Pattern: `(?!^[0-9]+$)([a-zA-Z0-9-_]+)`
- **description** - (Optional) Description of the alias.
- **function\_name** - (Required) The function ARN of the Lambda function for which you want to create an alias.
- **function\_version** - (Required) Lambda function version for which you are creating the alias. Pattern: `(\\$LATEST|[0-9]+)`.

### » **Attributes Reference**

- **arn** - The Amazon Resource Name (ARN) identifying your Lambda function alias.

## » **aws\_lambda\_event\_source\_mapping**

Provides a Lambda event source mapping. This allows Lambda functions to get events from Kinesis and DynamoDB.

For information about Lambda and how to use it, see [What is AWS Lambda?](#)  
For information about event source mappings, see [CreateEventSourceMapping](#) in the API docs.

## » Example Usage

```
resource "aws_lambda_event_source_mapping" "event_source_mapping" {
  batch_size      = 100
  event_source_arn = "arn:aws:kinesis:REGION:123456789012:stream/stream_name"
  enabled         = true
  function_name   = "arn:aws:lambda:REGION:123456789012:function:function_name"
  starting_position = "TRIM_HORIZON|LATEST"
}
```

## » Argument Reference

- **batch\_size** - (Optional) The largest number of records that Lambda will retrieve from your event source at the time of invocation. Defaults to 100.
- **event\_source\_arn** - (Required) The event source ARN - can either be a Kinesis or DynamoDB stream.
- **enabled** - (Optional) Determines if the mapping will be enabled on creation. Defaults to **true**.
- **function\_name** - (Required) The name or the ARN of the Lambda function that will be subscribing to events.
- **starting\_position** - (Required) The position in the stream where AWS Lambda should start reading. Can be one of either **TRIM\_HORIZON** or **LATEST**.

## » Attributes Reference

- **function\_arn** - The the ARN of the Lambda function the event source mapping is sending events to. (Note: this is a computed value that differs from **function\_name** above.)
- **last\_modified** - The date this resource was last modified.
- **last\_processing\_result** - The result of the last AWS Lambda invocation of your Lambda function.
- **state** - The state of the event source mapping.
- **state\_transition\_reason** - The reason the event source mapping is in its current state.
- **uuid** - The UUID of the created event source mapping.

## » Import

Lambda Event Source Mappings can be imported using the UUID (event source mapping identifier), e.g.

```
$ terraform import aws_lambda_event_source_mapping.event_source_mapping 12345kxodurf3443
```

## » aws\_\_lambda\_\_function

Provides a Lambda Function resource. Lambda allows you to trigger execution of code in response to events in AWS. The Lambda Function itself includes source code and runtime configuration.

For information about Lambda and how to use it, see [What is AWS Lambda?](#)

### » Example Usage

```
resource "aws_iam_role" "iam_for_lambda" {
  name = "iam_for_lambda"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "lambda.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
EOF
}

resource "aws_lambda_function" "test_lambda" {
  filename      = "lambda_function_payload.zip"
  function_name = "lambda_function_name"
  role          = "${aws_iam_role.iam_for_lambda.arn}"
  handler       = "exports.test"
  source_code_hash = "${base64sha256(file("lambda_function_payload.zip"))}"
  runtime       = "nodejs4.3"

  environment {
    variables = {
      foo = "bar"
    }
  }
}
```

## » Specifying the Deployment Package

AWS Lambda expects source code to be provided as a deployment package whose structure varies depending on which `runtime` is in use. See [Runtimes](#) for the valid values of `runtime`. The expected structure of the deployment package can be found in the [AWS Lambda documentation](#) for each runtime.

Once you have created your deployment package you can specify it either directly as a local file (using the `filename` argument) or indirectly via Amazon S3 (using the `s3_bucket`, `s3_key` and `s3_object_version` arguments). When providing the deployment package via S3 it may be useful to use the `aws_s3_bucket_object` resource to upload it.

For larger deployment packages it is recommended by Amazon to upload via S3, since the S3 API has better support for uploading large files efficiently.

## » Argument Reference

- `filename` - (Optional) The path to the function's deployment package within the local filesystem. If defined, The `s3_`-prefixed options cannot be used.
- `s3_bucket` - (Optional) The S3 bucket location containing the function's deployment package. Conflicts with `filename`. This bucket must reside in the same AWS region where you are creating the Lambda function.
- `s3_key` - (Optional) The S3 key of an object containing the function's deployment package. Conflicts with `filename`.
- `s3_object_version` - (Optional) The object version containing the function's deployment package. Conflicts with `filename`.
- `function_name` - (Required) A unique name for your Lambda Function.
- `dead_letter_config` - (Optional) Nested block to configure the function's *dead letter queue*. See details below.
- `handler` - (Required) The function entrypoint in your code.
- `role` - (Required) IAM role attached to the Lambda Function. This governs both who / what can invoke your Lambda Function, as well as what resources our Lambda Function has access to. See [Lambda Permission Model](#) for more details.
- `description` - (Optional) Description of what your Lambda Function does.
- `memory_size` - (Optional) Amount of memory in MB your Lambda Function can use at runtime. Defaults to 128. See [Limits](#)
- `runtime` - (Required) See [Runtimes](#) for valid values.
- `timeout` - (Optional) The amount of time your Lambda Function has to run in seconds. Defaults to 3. See [Limits](#)
- `reserved_concurrent_executions` - (Optional) The amount of reserved concurrent executions for this lambda function. Defaults to [Unreserved Concurrency Limits](#). See [Managing Concurrency](#)

- **publish** - (Optional) Whether to publish creation/change as new Lambda Function Version. Defaults to **false**.
- **vpc\_config** - (Optional) Provide this to allow your function to access your VPC. Fields documented below. See Lambda in VPC
- **environment** - (Optional) The Lambda environment's configuration settings. Fields documented below.
- **kms\_key\_arn** - (Optional) The ARN for the KMS encryption key.
- **source\_code\_hash** - (Optional) Used to trigger updates. Must be set to a base64-encoded SHA256 hash of the package file specified with either **filename** or **s3\_key**. The usual way to set this is `${base64sha256(file("file.zip"))}`, where "file.zip" is the local filename of the lambda function source archive.
- **tags** - (Optional) A mapping of tags to assign to the object.

**dead\_letter\_config** is a child block with a single argument:

- **target\_arn** - (Required) The ARN of an SNS topic or SQS queue to notify when an invocation fails. If this option is used, the function's IAM role must be granted suitable access to write to the target object, which means allowing either the **sns:Publish** or **sqs:SendMessage** action on this ARN, depending on which service is targeted.

**tracing\_config** is a child block with a single argument:

- **mode** - (Required) Can be either **PassThrough** or **Active**. If **PassThrough**, Lambda will only trace the request from an upstream service if it contains a tracing header with "sampled=1". If **Active**, Lambda will respect any tracing header it receives from an upstream service. If no tracing header is received, Lambda will call X-Ray for a tracing decision.

**vpc\_config** requires the following:

- **subnet\_ids** - (Required) A list of subnet IDs associated with the Lambda function.
- **security\_group\_ids** - (Required) A list of security group IDs associated with the Lambda function.

**NOTE:** if both **subnet\_ids** and **security\_group\_ids** are empty then **vpc\_config** is considered to be empty or unset.

For **environment** the following attributes are supported:

- **variables** - (Optional) A map that defines environment variables for the Lambda function.

## » Attributes Reference

- **arn** - The Amazon Resource Name (ARN) identifying your Lambda Function.



- `qualified_arn` - The Amazon Resource Name (ARN) identifying your Lambda Function Version (if versioning is enabled via `publish = true`).
- `invoke_arn` - The ARN to be used for invoking Lambda Function from API Gateway - to be used in `aws_api_gateway_integration`'s `uri`
- `version` - Latest published version of your Lambda Function.
- `last_modified` - The date this resource was last modified.
- `kms_key_arn` - (Optional) The ARN for the KMS encryption key.
- `source_code_hash` - Base64-encoded representation of raw SHA-256 sum of the zip file, provided either via `filename` or `s3_*` parameters.
- `source_code_size` - The size in bytes of the function .zip file.

## » Import

Lambda Functions can be imported using the `function_name`, e.g.

```
$ terraform import aws_lambda_function.test_lambda my_test_lambda_function
```

## » `aws_lambda_permission`

Creates a Lambda permission to allow external sources invoking the Lambda function (e.g. CloudWatch Event Rule, SNS or S3).

## » Example Usage

```
resource "aws_lambda_permission" "allow_cloudwatch" {
  statement_id  = "AllowExecutionFromCloudWatch"
  action       = "lambda:InvokeFunction"
  function_name = "${aws_lambda_function.test_lambda.function_name}"
  principal    = "events.amazonaws.com"
  source_arn    = "arn:aws:events:eu-west-1:111122223333:rule/RunDaily"
  qualifier    = "${aws_lambda_alias.test_alias.name}"
}

resource "aws_lambda_alias" "test_alias" {
  name           = "testalias"
  description    = "a sample description"
  function_name  = "${aws_lambda_function.test_lambda.function_name}"
  function_version = "$LATEST"
}

resource "aws_lambda_function" "test_lambda" {
  filename      = "lambdatest.zip"
  function_name = "lambda_function_name"
}
```

```

    role          = "${aws_iam_role.iam_for_lambda.arn}"
    handler       = "exports.handler"
    runtime       = "nodejs6.10"
  }

  resource "aws_iam_role" "iam_for_lambda" {
    name = "iam_for_lambda"

    assume_role_policy = <<EOF
  {
    "Version": "2012-10-17",
    "Statement": [
      {
        "Action": "sts:AssumeRole",
        "Principal": {
          "Service": "lambda.amazonaws.com"
        },
        "Effect": "Allow",
        "Sid": ""
      }
    ]
  }
  }
EOF
}

```

## » Usage with SNS

```

resource "aws_lambda_permission" "with_sns" {
  statement_id = "AllowExecutionFromSNS"
  action      = "lambda:InvokeFunction"
  function_name = "${aws_lambda_function.func.function_name}"
  principal    = "sns.amazonaws.com"
  source_arn    = "${aws_sns_topic.default.arn}"
}

resource "aws_sns_topic" "default" {
  name = "call-lambda-maybe"
}

resource "aws_sns_topic_subscription" "lambda" {
  topic_arn = "${aws_sns_topic.default.arn}"
  protocol  = "lambda"
  endpoint  = "${aws_lambda_function.func.arn}"
}

```

```

resource "aws_lambda_function" "func" {
  filename      = "lambdatest.zip"
  function_name = "lambda_called_from_sns"
  role          = "${aws_iam_role.default.arn}"
  handler       = "exports.handler"
  runtime       = "python2.7"
}

resource "aws_iam_role" "default" {
  name = "iam_for_lambda_with_sns"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "lambda.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
EOF
}

```

## » Argument Reference

- **action** - (Required) The AWS Lambda action you want to allow in this statement. (e.g. `lambda:InvokeFunction`)
- **function\_name** - (Required) Name of the Lambda function whose resource policy you are updating
- **principal** - (Required) The principal who is getting this permission. e.g. `s3.amazonaws.com`, an AWS account ID, or any valid AWS service principal such as `events.amazonaws.com` or `sns.amazonaws.com`.
- **qualifier** - (Optional) Query parameter to specify function version or alias name. The permission will then apply to the specific qualified ARN. e.g. `arn:aws:lambda:aws-region:acct-id:function:function-name:2`
- **source\_account** - (Optional) This parameter is used for S3 and SES. The AWS account ID (without a hyphen) of the source owner.
- **source\_arn** - (Optional) When granting Amazon S3 or CloudWatch Events permission to invoke your function, you should specify this field with the Amazon Resource Name (ARN) for the S3 Bucket or

CloudWatch Events Rule as its value. This ensures that only events generated from the specified bucket or rule can invoke the function. API Gateway ARNs have a unique structure described here.

- **statement\_id** - (Optional) A unique statement identifier. By default generated by Terraform.
- **statement\_id\_prefix** - (Optional) A statement identifier prefix. Terraform will generate a unique suffix. Conflicts with **statement\_id**.

## » **aws\_\_lightsail\_\_domain**

Creates a domain resource for the specified domain (e.g., example.com). You cannot register a new domain name using Lightsail. You must register a domain name using Amazon Route 53 or another domain name registrar. If you have already registered your domain, you can enter its name in this parameter to manage the DNS records for that domain.

**Note:** Lightsail is currently only supported in a limited number of AWS Regions, please see "Regions and Availability Zones in Amazon Lightsail" for more details

### » **Example Usage, creating a new domain**

```
resource "aws_lightsail_domain" "domain_test" {  
  domain_name = "mydomain.com"  
}
```

### » **Argument Reference**

The following arguments are supported:

- **domain\_name** - (Required) The name of the Lightsail domain to manage

### » **Attributes Reference**

The following attributes are exported in addition to the arguments listed above:

- **id** - The name used for this domain
- **arn** - The ARN of the Lightsail domain

## » **aws\_\_lightsail\_\_instance**

Provides a Lightsail Instance. Amazon Lightsail is a service to provide easy virtual private servers with custom software already setup. See What is Amazon

Lightsail? for more information.

**Note:** Lightsail is currently only supported in a limited number of AWS Regions, please see "Regions and Availability Zones in Amazon Lightsail" for more details

## » Example Usage

```
# Create a new GitLab Lightsail Instance
resource "aws_lightsail_instance" "gitlab_test" {
  name           = "custom gitlab"
  availability_zone = "us-east-1b"
  blueprint_id    = "string"
  bundle_id       = "string"
  key_pair_name   = "some_key_name"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the Lightsail Instance
- **availability\_zone** - (Required) The Availability Zone in which to create your instance. At this time, must be in `us-east-1`, `us-east-2`, `us-west-2`, `eu-west-1`, `eu-west-2`, `eu-central-1`, `ap-southeast-1`, `ap-southeast-2`, `ap-northeast-1`, `ap-south-1` regions
- **blueprint\_id** - (Required) The ID for a virtual private server image (see list below)
- **bundle\_id** - (Required) The bundle of specification information (see list below)
- **key\_pair\_name** - (Required) The name of your key pair. Created in the Lightsail console (cannot use `aws_key_pair` at this time)
- **user\_data** - (Optional) launch script to configure server with additional user data

## » Blueprints

Lightsail currently supports the following Blueprint IDs: - `amazon_linux_2017_03_1_1` - `ubuntu_16_04_1` - `debian_8_7` - `freebsd_11` - `opensuse_42_2` - `wordpress_4_8_0` - `lamp_5_6_30_5` - `nodejs_7_10_0` - `joomla_3_7_3` - `magento_2_1_7` - `mean_3_4_5` - `drupal_8_3_3` - `gitlab_9_2_6` - `redmine_3_3_3_1` - `nginx_1_12_0_2`

## » Bundles

Lightsail currently supports the following Bundle IDs:

- nano\_1\_0
- micro\_1\_0
- small\_1\_0
- medium\_1\_0
- large\_1\_0

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- `id` - The ARN of the Lightsail instance (matches `arn`).
- `arn` - The ARN of the Lightsail instance (matches `id`).
- `availability_zone`
- `blueprint_id`
- `bundle_id`
- `key_pair_name`
- `user_data`

## » Import

Lightsail Instances can be imported using their ARN, e.g.

```
$ terraform import aws_lightsail_instance.bar <arn>
```

## » `aws_lightsail_key_pair`

Provides a Lightsail Key Pair, for use with Lightsail Instances. These key pairs are separate from EC2 Key Pairs, and must be created or imported for use with Lightsail.

**Note:** Lightsail is currently only supported in a limited number of AWS Regions, please see "Regions and Availability Zones in Amazon Lightsail" for more details

## » Example Usage, creating a new Key Pair

```
# Create a new Lightsail Key Pair
resource "aws_lightsail_key_pair" "lg_key_pair" {
  name = "lg_key_pair"
}
```

## » Create new Key Pair, encrypting the private key with a PGP Key

```
resource "awsLightsailKeyPair" "lg_key_pair" {
  name      = "lg_key_pair"
  pgp_key   = "keybase:keybaseusername"
}
```

## » Import an existing public key

```
resource "awsLightsailKeyPair" "lg_key_pair" {
  name      = "importing"
  public_key = "${file("~/ssh/id_rsa.pub")}"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The name of the Lightsail Key Pair. If omitted, a unique name will be generated by Terraform
- **pgp\_key** - (Optional) An optional PGP key to encrypt the resulting private key material. Only used when creating a new key pair
- **public\_key** - (Required) The public key material. This public key will be imported into Lightsail

**NOTE:** a PGP key is not required, however it is strongly encouraged. Without a PGP key, the private key material will be stored in state unencrypted. **pgp\_key** is ignored if **public\_key** is supplied.

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- **id** - The name used for this key pair
- **arn** - The ARN of the Lightsail key pair
- **fingerprint** - The MD5 public key fingerprint as specified in section 4 of RFC 4716.
- **public\_key** - the public key, base64 encoded
- **private\_key** - the private key, base64 encoded. This is only populated when creating a new key, and when no **pgp\_key** is provided
- **encrypted\_private\_key** - the private key material, base 64 encoded and encrypted with the given **pgp\_key**. This is only populated when creating a new key and **pgp\_key** is supplied

- **encrypted\_fingerprint** - The MD5 public key fingerprint for the encrypted private key

## » Import

Lightsail Key Pairs cannot be imported, because the private and public key are only available on initial creation.

## » **aws\_\_lightsail\_\_static\_\_ip**

Allocates a static IP address.

**Note:** Lightsail is currently only supported in a limited number of AWS Regions, please see "Regions and Availability Zones in Amazon Lightsail" for more details

## » Example Usage

```
resource "aws_lightsail_static_ip" "test" {
  name = "example"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name for the allocated static IP

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- **arn** - The ARN of the Lightsail static IP
- **ip\_address** - The allocated static IP address
- **support\_code** - The support code.

## » **aws\_\_lightsail\_\_static\_\_ip\_\_attachment**

Provides a static IP address attachment - relationship between a Lightsail static IP & Lightsail instance.



**Note:** Lightsail is currently only supported in a limited number of AWS Regions, please see "Regions and Availability Zones in Amazon Lightsail" for more details

## » Example Usage

```
resource "aws_lightsail_static_ip_attachment" "test" {
  static_ip_name = "${aws_lightsail_static_ip.test.name}"
  instance_name = "${aws_lightsail_instance.test.name}"
}

resource "aws_lightsail_static_ip" "test" {
  name = "example"
}

resource "aws_lightsail_instance" "test" {
  name                = "example"
  availability_zone   = "us-east-1b"
  blueprint_id        = "string"
  bundle_id           = "string"
  key_pair_name       = "some_key_name"
}
```

## » Argument Reference

The following arguments are supported:

- `static_ip_name` - (Required) The name of the allocated static IP
- `instance_name` - (Required) The name of the Lightsail instance to attach the IP to

## » Attributes Reference

The following attributes are exported in addition to the arguments listed above:

- `arn` - The ARN of the Lightsail static IP
- `ip_address` - The allocated static IP address
- `support_code` - The support code.

## » `aws__mq__broker`

Provides an MQ Broker Resource. This resources also manages users for the broker.

For more information on Amazon MQ, see Amazon MQ documentation.

Changes to an MQ Broker can occur when you change a parameter, such as `configuration` or `user`, and are reflected in the next maintenance window. Because of this, Terraform may report a difference in its planning phase because a modification has not yet taken place. You can use the `apply_immediately` flag to instruct the service to apply the change immediately (see documentation below).

**Note:** using `apply_immediately` can result in a brief downtime as the broker reboots.

**Note:** All arguments including the username and password will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Example Usage

```
resource "aws_mq_broker" "example" {
  broker_name = "example"
  configuration {
    id = "${aws_mq_configuration.test.id}"
    revision = "${aws_mq_configuration.test.latest_revision}"
  }
  engine_type = "ActiveMQ"
  engine_version = "5.15.0"
  host_instance_type = "mq.t2.micro"
  security_groups = ["${aws_security_group.test.id}"]
  user {
    username = "ExampleUser"
    password = "MindTheGap"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `apply_immediately` - (Optional) Specifies whether any broker modifications are applied immediately, or during the next maintenance window. Default is `false`.
- `auto_minor_version_upgrade` - (Optional) Enables automatic upgrades to new minor versions for brokers, as Apache releases the versions.
- `broker_name` - (Required) The name of the broker.
- `configuration` - (Optional) Configuration of the broker. See below.

- **deployment\_mode** - (Optional) The deployment mode of the broker. Supported: `SINGLE_INSTANCE` and `ACTIVE_STANDBY_MULTI_AZ`. Defaults to `SINGLE_INSTANCE`.
- **engine\_type** - (Required) The type of broker engine. Currently, Amazon MQ supports only `ActiveMQ`.
- **engine\_version** - (Required) The version of the broker engine. Currently, Amazon MQ supports only `5.15.0`.
- **host\_instance\_type** - (Required) The broker's instance type. e.g. `mq.t2.micro` or `mq.m4.large`
- **publicly\_accessible** - (Optional) Whether to enable connections from applications outside of the VPC that hosts the broker's subnets.
- **security\_groups** - (Required) The list of security group IDs assigned to the broker.
- **subnet\_ids** - (Optional) The list of subnet IDs in which to launch the broker. A `SINGLE_INSTANCE` deployment requires one subnet. An `ACTIVE_STANDBY_MULTI_AZ` deployment requires two subnets.
- **maintenance\_window\_start\_time** - (Optional) Maintenance window start time. See below.
- **user** - (Optional) The list of all ActiveMQ usernames for the specified broker. See below.

## » Nested Fields

### » **configuration**

- **id** - (Optional) The Configuration ID.
- **revision** - (Optional) Revision of the Configuration.

### » **maintenance\_window\_start\_time**

- **day\_of\_week** - (Required) The day of the week. e.g. `MONDAY`, `TUESDAY`, or `WEDNESDAY`
- **time\_of\_day** - (Required) The time, in 24-hour format. e.g. `02:00`
- **time\_zone** - (Required) The time zone, UTC by default, in either the Country/City format, or the UTC offset format. e.g. `CET`

### » **user**

- **console\_access** - (Optional) Whether to enable access to the the ActiveMQ Web Console for the user.
- **groups** - (Optional) The list of groups (20 maximum) to which the ActiveMQ user belongs.
- **password** - (Required) The password of the user. It must be 12 to 250 characters long, at least 4 unique characters, and must not contain commas.

- **username** - (Required) The username of the user.

## » Attributes Reference

The following attributes are exported:

- **id** - The unique ID that Amazon MQ generates for the broker.
- **arn** - The ARN of the broker.
- **instances** - A list of information about allocated brokers (both active & standby).
  - **instances.0.console\_url** - The URL of the broker's ActiveMQ Web Console.
  - **instances.0.endpoints** - The broker's wire-level protocol endpoints in the following order & format referenceable e.g. as **instances.0.endpoints.0 (SSL)**:
    - \* **ssl**://broker-id.mq.us-west-2.amazonaws.com:61617
    - \* **amqp+ssl**://broker-id.mq.us-west-2.amazonaws.com:5671
    - \* **stomp+ssl**://broker-id.mq.us-west-2.amazonaws.com:61614
    - \* **mqtt+ssl**://broker-id.mq.us-west-2.amazonaws.com:8883
    - \* **wss**://broker-id.mq.us-west-2.amazonaws.com:61619

## » Import

MQ Broker is currently not importable.

## » aws\_\_mq\_\_configuration

Provides an MQ Configuration Resource.

For more information on Amazon MQ, see Amazon MQ documentation.

## » Example Usage

```
resource "aws_mq_configuration" "example" {
  description      = "Example Configuration"
  name             = "example"
  engine_type      = "ActiveMQ"
  engine_version   = "5.15.0"
  data             = <<DATA
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<broker xmlns="http://activemq.apache.org/schema/core">
  <plugins>
```

```

        <forcePersistencyModeBrokerPlugin persistenceFlag="true"/>
        <statisticsBrokerPlugin/>
        <timeStampingBrokerPlugin ttlCeiling="86400000" zeroExpirationOverride="86400000"/>
    </plugins>
</broker>
DATA
}
```

## » Argument Reference

The following arguments are supported:

- **data** - (Required) The broker configuration in XML format. See official docs for supported parameters and format of the XML.
- **description** - (Optional) The description of the configuration.
- **engine\_type** - (Required) The type of broker engine.
- **engine\_version** - (Required) The version of the broker engine.
- **name** - (Required) The name of the configuration

## » Attributes Reference

The following attributes are exported:

- **id** - The unique ID that Amazon MQ generates for the configuration.
- **arn** - The ARN of the configuration.
- **latest\_revision** - The latest revision of the configuration.

## » Import

MQ Configurations can be imported using the configuration ID, e.g.

```
$ terraform import aws_mq_configuration.example c-0187d1eb-88c8-475a-9b79-16ef5a10c94f
```

## » aws\_\_media\_\_store\_\_container

Provides a MediaStore Container.

## » Example Usage

```
resource "aws_media_store_container" "example" {
  name = "example"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the container. Must contain alphanumeric characters or underscores.

## » Attributes Reference

The following attributes are exported:

- **arn** - The ARN of the container.
- **endpoint** - The DNS endpoint of the container.

## » `aws_opsworks_application`

Provides an OpsWorks application resource.

## » Example Usage

```
resource "aws_opsworks_application" "foo-app" {
  name          = "foobar application"
  short_name    = "foobar"
  stack_id      = "${aws_opsworks_stack.stack.id}"
  type          = "rails"
  description   = "This is a Rails application"

  domains = [
    "example.com",
    "sub.example.com",
  ]

  environment = {
    key   = "key"
    value = "value"
    secure = false
  }

  app_source = {
    type       = "git"
    revision   = "master"
    url        = "https://github.com/example.git"
  }
}
```

```

enable_ssl = true

ssl_configuration = {
  private_key = "${file("./foobar.key")}"
  certificate = "${file("./foobar.crt")}"
}

document_root      = "public"
auto_bundle_on_deploy = true
rails_env          = "staging"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A human-readable name for the application.
- **short\_name** - (Required) A short, machine-readable name for the application. This can only be defined on resource creation and ignored on resource update.
- **stack\_id** - (Required) The id of the stack the application will belong to.
- **type** - (Required) Opsworks application type. One of `aws-flow-ruby`, `java`, `rails`, `php`, `nodejs`, `static` or `other`.
- **description** - (Optional) A description of the app.
- **environment** - (Optional) Object to define environment variables. Object is described below.
- **enable\_ssl** - (Optional) Whether to enable SSL for the app. This must be set in order to let `ssl_configuration.private_key`, `ssl_configuration.certificate` and `ssl_configuration.chain` take effect.
- **ssl\_configuration** - (Optional) The SSL configuration of the app. Object is described below.
- **app\_source** - (Optional) SCM configuration of the app as described below.
- **data\_source\_arn** - (Optional) The data source's ARN.
- **data\_source\_type** - (Optional) The data source's type one of `AutoSelectOpsworksMySQLInstance`, `OpsworksMySQLInstance`, or `RdsDbInstance`.
- **data\_source\_database\_name** - (Optional) The database name.
- **domains** - (Optional) A list of virtual host alias.
- **document\_root** - (Optional) Subfolder for the document root for application of type `rails`.
- **auto\_bundle\_on\_deploy** - (Optional) Run bundle install when deploying for application of type `rails`.
- **rails\_env** - (Required if `type = rails`) The name of the Rails environ-

ment for application of type `rails`.

- `aws_flow_ruby_settings` - (Optional) Specify activity and workflow workers for your app using the `aws-flow` gem.

An `app_source` block supports the following arguments (can only be defined once per resource):

- `type` - (Required) The type of source to use. For example, "archive".
- `url` - (Required) The URL where the app resource can be found.
- `username` - (Optional) Username to use when authenticating to the source.
- `password` - (Optional) Password to use when authenticating to the source.
- `ssh_key` - (Optional) SSH key to use when authenticating to the source.
- `revision` - (Optional) For sources that are version-aware, the revision to use.

An `environment` block supports the following arguments:

- `key` - (Required) Variable name.
- `value` - (Required) Variable value.
- `secure` - (Optional) Set visibility of the variable value to `true` or `false`.

A `ssl_configuration` block supports the following arguments (can only be defined once per resource):

- `private_key` - (Required) The private key; the contents of the certificate's domain.key file.
- `certificate` - (Required) The contents of the certificate's domain.crt file.
- `chain` - (Optional) Can be used to specify an intermediate certificate authority key or client authentication.

## » Attributes Reference

The following attributes are exported:

- `id` - The id of the application.

## » `aws_opsworks__custom__layer`

Provides an OpsWorks custom layer resource.

## » Example Usage

```
resource "aws_opsworks_custom_layer" "custlayer" {
  name          = "My Awesome Custom Layer"
  short_name    = "awesome"
  stack_id      = "${aws_opsworks_stack.main.id}"
```



}

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A human-readable name for the layer.
- **short\_name** - (Required) A short, machine-readable name for the layer, which will be used to identify it in the Chef node JSON.
- **stack\_id** - (Required) The id of the stack the layer will belong to.
- **auto\_assign\_elastic\_ips** - (Optional) Whether to automatically assign an elastic IP address to the layer's instances.
- **auto\_assign\_public\_ips** - (Optional) For stacks belonging to a VPC, whether to automatically assign a public IP address to each of the layer's instances.
- **custom\_instance\_profile\_arn** - (Optional) The ARN of an IAM profile that will be used for the layer's instances.
- **custom\_security\_group\_ids** - (Optional) Ids for a set of security groups to apply to the layer's instances.
- **auto\_healing** - (Optional) Whether to enable auto-healing for the layer.
- **install\_updates\_on\_boot** - (Optional) Whether to install OS and package updates on each instance when it boots.
- **instance\_shutdown\_timeout** - (Optional) The time, in seconds, that OpsWorks will wait for Chef to complete after triggering the Shutdown event.
- **elastic\_load\_balancer** - (Optional) Name of an Elastic Load Balancer to attach to this layer
- **drain\_elb\_on\_shutdown** - (Optional) Whether to enable Elastic Load Balancing connection draining.
- **system\_packages** - (Optional) Names of a set of system packages to install on the layer's instances.
- **use\_ebs\_optimized\_instances** - (Optional) Whether to use EBS-optimized instances.
- **ebs\_volume** - (Optional) **ebs\_volume** blocks, as described below, will each create an EBS volume and connect it to the layer's instances.
- **custom\_json** - (Optional) Custom JSON attributes to apply to the layer.

The following extra optional arguments, all lists of Chef recipe names, allow custom Chef recipes to be applied to layer instances at the five different lifecycle events, if custom cookbooks are enabled on the layer's stack:

- **custom\_configure\_recipes**
- **custom\_deploy\_recipes**
- **custom\_setup\_recipes**
- **custom\_shutdown\_recipes**
- **custom\_undeploy\_recipes**

An `ebs_volume` block supports the following arguments:

- `mount_point` - (Required) The path to mount the EBS volume on the layer's instances.
- `size` - (Required) The size of the volume in gigabytes.
- `number_of_disks` - (Required) The number of disks to use for the EBS volume.
- `raid_level` - (Required) The RAID level to use for the volume.
- `type` - (Optional) The type of volume to create. This may be `standard` (the default), `io1` or `gp2`.
- `iops` - (Optional) For PIOPS volumes, the IOPS per disk.

## » Attributes Reference

The following attributes are exported:

- `id` - The id of the layer.

## » Import

OpsWorks Custom Layers can be imported using the `id`, e.g.

```
$ terraform import aws_opsworks_custom_layer.bar 00000000-0000-0000-0000-000000000000
```

## » `aws__opsworks__ganglia__layer`

Provides an OpsWorks Ganglia layer resource.

## » Example Usage

```
resource "aws_opsworks_ganglia_layer" "monitor" {
  stack_id = "${aws_opsworks_stack.main.id}"
  password = "foobarbaz"
}
```

## » Argument Reference

The following arguments are supported:

- `stack_id` - (Required) The id of the stack the layer will belong to.
- `password` - (Required) The password to use for Ganglia.
- `name` - (Optional) A human-readable name for the layer.

- **auto\_assign\_elastic\_ips** - (Optional) Whether to automatically assign an elastic IP address to the layer's instances.
- **auto\_assign\_public\_ips** - (Optional) For stacks belonging to a VPC, whether to automatically assign a public IP address to each of the layer's instances.
- **custom\_instance\_profile\_arn** - (Optional) The ARN of an IAM profile that will be used for the layer's instances.
- **custom\_security\_group\_ids** - (Optional) Ids for a set of security groups to apply to the layer's instances.
- **auto\_healing** - (Optional) Whether to enable auto-healing for the layer.
- **install\_updates\_on\_boot** - (Optional) Whether to install OS and package updates on each instance when it boots.
- **instance\_shutdown\_timeout** - (Optional) The time, in seconds, that OpsWorks will wait for Chef to complete after triggering the Shutdown event.
- **elastic\_load\_balancer** - (Optional) Name of an Elastic Load Balancer to attach to this layer
- **drain\_elb\_on\_shutdown** - (Optional) Whether to enable Elastic Load Balancing connection draining.
- **system\_packages** - (Optional) Names of a set of system packages to install on the layer's instances.
- **url** - (Optional) The URL path to use for Ganglia. Defaults to `"/ganglia"`.
- **username** - (Optional) The username to use for Ganglia. Defaults to `"opsworks"`.
- **use\_ebs\_optimized\_instances** - (Optional) Whether to use EBS-optimized instances.
- **ebs\_volume** - (Optional) **ebs\_volume** blocks, as described below, will each create an EBS volume and connect it to the layer's instances.
- **custom\_json** - (Optional) Custom JSON attributes to apply to the layer.

The following extra optional arguments, all lists of Chef recipe names, allow custom Chef recipes to be applied to layer instances at the five different lifecycle events, if custom cookbooks are enabled on the layer's stack:

- **custom\_configure\_recipes**
- **custom\_deploy\_recipes**
- **custom\_setup\_recipes**
- **custom\_shutdown\_recipes**
- **custom\_undeploy\_recipes**

An **ebs\_volume** block supports the following arguments:

- **mount\_point** - (Required) The path to mount the EBS volume on the layer's instances.
- **size** - (Required) The size of the volume in gigabytes.
- **number\_of\_disks** - (Required) The number of disks to use for the EBS volume.
- **raid\_level** - (Required) The RAID level to use for the volume.

- **type** - (Optional) The type of volume to create. This may be **standard** (the default), **io1** or **gp2**.
- **iops** - (Optional) For PIOPS volumes, the IOPS per disk.

## » Attributes Reference

The following attributes are exported:

- **id** - The id of the layer.

## » `aws_opsworks_haproxy_layer`

Provides an OpsWorks haproxy layer resource.

## » Example Usage

```
resource "aws_opsworks_haproxy_layer" "lb" {
  stack_id      = "${aws_opsworks_stack.main.id}"
  stats_password = "foobazbar"
}
```

## » Argument Reference

The following arguments are supported:

- **stack\_id** - (Required) The id of the stack the layer will belong to.
- **stats\_password** - (Required) The password to use for HAProxy stats.
- **name** - (Optional) A human-readable name for the layer.
- **auto\_assign\_elastic\_ips** - (Optional) Whether to automatically assign an elastic IP address to the layer's instances.
- **auto\_assign\_public\_ips** - (Optional) For stacks belonging to a VPC, whether to automatically assign a public IP address to each of the layer's instances.
- **custom\_instance\_profile\_arn** - (Optional) The ARN of an IAM profile that will be used for the layer's instances.
- **custom\_security\_group\_ids** - (Optional) Ids for a set of security groups to apply to the layer's instances.
- **auto\_healing** - (Optional) Whether to enable auto-healing for the layer.
- **healthcheck\_method** - (Optional) HTTP method to use for instance healthchecks. Defaults to "OPTIONS".
- **healthcheck\_url** - (Optional) URL path to use for instance healthchecks. Defaults to "/".

- `install_updates_on_boot` - (Optional) Whether to install OS and package updates on each instance when it boots.
- `instance_shutdown_timeout` - (Optional) The time, in seconds, that OpsWorks will wait for Chef to complete after triggering the Shutdown event.
- `elastic_load_balancer` - (Optional) Name of an Elastic Load Balancer to attach to this layer
- `drain_elb_on_shutdown` - (Optional) Whether to enable Elastic Load Balancing connection draining.
- `stats_enabled` - (Optional) Whether to enable HAProxy stats.
- `stats_url` - (Optional) The HAProxy stats URL. Defaults to `"/haproxy?stats"`.
- `stats_user` - (Optional) The username for HAProxy stats. Defaults to `"opsworks"`.
- `system_packages` - (Optional) Names of a set of system packages to install on the layer's instances.
- `use_ebs_optimized_instances` - (Optional) Whether to use EBS-optimized instances.
- `ebs_volume` - (Optional) `ebs_volume` blocks, as described below, will each create an EBS volume and connect it to the layer's instances.
- `custom_json` - (Optional) Custom JSON attributes to apply to the layer.

The following extra optional arguments, all lists of Chef recipe names, allow custom Chef recipes to be applied to layer instances at the five different lifecycle events, if custom cookbooks are enabled on the layer's stack:

- `custom_configure_recipes`
- `custom_deploy_recipes`
- `custom_setup_recipes`
- `custom_shutdown_recipes`
- `custom_undeploy_recipes`

An `ebs_volume` block supports the following arguments:

- `mount_point` - (Required) The path to mount the EBS volume on the layer's instances.
- `size` - (Required) The size of the volume in gigabytes.
- `number_of_disks` - (Required) The number of disks to use for the EBS volume.
- `raid_level` - (Required) The RAID level to use for the volume.
- `type` - (Optional) The type of volume to create. This may be `standard` (the default), `io1` or `gp2`.
- `iops` - (Optional) For PIOPS volumes, the IOPS per disk.

## » Attributes Reference

The following attributes are exported:

- `id` - The id of the layer.

## » `aws__opsworks__instance`

Provides an OpsWorks instance resource.

### » Example Usage

```
resource "aws_opsworks_instance" "my-instance" {
  stack_id = "${aws_opsworks_stack.my-stack.id}"

  layer_ids = [
    "${aws_opsworks_custom_layer.my-layer.id}",
  ]

  instance_type = "t2.micro"
  os             = "Amazon Linux 2015.09"
  state         = "stopped"
}
```

### » Argument Reference

The following arguments are supported:

- `instance_type` - (Required) The type of instance to start
- `stack_id` - (Required) The id of the stack the instance will belong to.
- `layer_ids` - (Required) The ids of the layers the instance will belong to.
- `state` - (Optional) The desired state of the instance. Can be either "running" or "stopped".
- `install_updates_on_boot` - (Optional) Controls where to install OS and package updates when the instance boots. Defaults to `true`.
- `auto_scaling_type` - (Optional) Creates load-based or time-based instances. If set, can be either: "load" or "timer".
- `availability_zone` - (Optional) Name of the availability zone where instances will be created by default.
- `ebs_optimized` - (Optional) If true, the launched EC2 instance will be EBS-optimized.
- `hostname` - (Optional) The instance's host name.
- `architecture` - (Optional) Machine architecture for created instances. Can be either "x86\_64" (the default) or "i386"
- `ami_id` - (Optional) The AMI to use for the instance. If an AMI is specified, `os` must be "Custom".
- `os` - (Optional) Name of operating system that will be installed.

- `root_device_type` - (Optional) Name of the type of root device instances will have by default. Can be either `"ebs"` or `"instance-store"`
- `ssh_key_name` - (Optional) Name of the SSH keypair that instances will have by default.
- `agent_version` - (Optional) The AWS OpsWorks agent to install. Defaults to `"INHERIT"`.
- `subnet_id` - (Optional) Subnet ID to attach to
- `tenancy` - (Optional) Instance tenancy to use. Can be one of `"default"`, `"dedicated"` or `"host"`
- `virtualization_type` - (Optional) Keyword to choose what virtualization mode created instances will use. Can be either `"paravirtual"` or `"hvm"`.
- `root_block_device` - (Optional) Customize details about the root block device of the instance. See Block Devices below for details.
- `ebs_block_device` - (Optional) Additional EBS block devices to attach to the instance. See Block Devices below for details.
- `ephemeral_block_device` - (Optional) Customize Ephemeral (also known as "Instance Store") volumes on the instance. See Block Devices below for details.

## » Block devices

Each of the `*_block_device` attributes controls a portion of the AWS Instance's "Block Device Mapping". It's a good idea to familiarize yourself with AWS's Block Device Mapping docs to understand the implications of using these attributes.

The `root_block_device` mapping supports the following:

- `volume_type` - (Optional) The type of volume. Can be `"standard"`, `"gp2"`, or `"io1"`. (Default: `"standard"`).
- `volume_size` - (Optional) The size of the volume in gigabytes.
- `iops` - (Optional) The amount of provisioned IOPS. This must be set with a `volume_type` of `"io1"`.
- `delete_on_termination` - (Optional) Whether the volume should be destroyed on instance termination (Default: `true`).

Modifying any of the `root_block_device` settings requires resource replacement.

Each `ebs_block_device` supports the following:

- `device_name` - The name of the device to mount.
- `snapshot_id` - (Optional) The Snapshot ID to mount.
- `volume_type` - (Optional) The type of volume. Can be `"standard"`, `"gp2"`, or `"io1"`. (Default: `"standard"`).
- `volume_size` - (Optional) The size of the volume in gigabytes.

- `iops` - (Optional) The amount of provisioned IOPS. This must be set with a `volume_type` of `"io1"`.
- `delete_on_termination` - (Optional) Whether the volume should be destroyed on instance termination (Default: `true`).

Modifying any `ebs_block_device` currently requires resource replacement.

Each `ephemeral_block_device` supports the following:

- `device_name` - The name of the block device to mount on the instance.
- `virtual_name` - The Instance Store Device Name (e.g. `"ephemeral0"`)

Each AWS Instance type has a different set of Instance Store block devices available for attachment. AWS publishes a list of which ephemeral devices are available on each type. The devices are always identified by the `virtual_name` in the format `"ephemeral{0..N}"`.

**NOTE:** Currently, changes to `*_block_device` configuration of *existing* resources cannot be automatically detected by Terraform. After making updates to block device configuration, resource recreation can be manually triggered by using the `taint` command.

## » Attributes Reference

The following attributes are exported:

- `id` - The id of the OpsWorks instance.
- `agent_version` - The AWS OpsWorks agent version.
- `availability_zone` - The availability zone of the instance.
- `ec2_instance_id` - EC2 instance ID
- `ssh_key_name` - The key name of the instance
- `public_dns` - The public DNS name assigned to the instance. For EC2-VPC, this is only available if you've enabled DNS hostnames for your VPC
- `public_ip` - The public IP address assigned to the instance, if applicable.
- `private_dns` - The private DNS name assigned to the instance. Can only be used inside the Amazon EC2, and only available if you've enabled DNS hostnames for your VPC
- `private_ip` - The private IP address assigned to the instance
- `subnet_id` - The VPC subnet ID.
- `tenancy` - The Instance tenancy
- `security_group_ids` - The associated security groups.

## » Timeouts

`aws_opsworks_instance` provides the following Timeouts configuration options:



- **create** - (Default 10 minutes) Used when the instance is created. It should cover the time needed for the instance to start successfully.
- **delete** - (Default 10 minutes) Used when the instance is deleted. It should cover the time needed for the instance to stop successfully.
- **update** - (Default 10 minutes) Used when the instance is changed. It should cover the time needed to either start or stop the instance.

## » Import

Opsworks Instances can be imported using the `instance_id`, e.g.

```
$ terraform import aws_opsworks_instance.my_instance 4d6d1710-ded9-42a1-b08e-b043ad7af1e2
```

## » aws\_opsworks\_java\_app\_layer

Provides an OpsWorks Java application layer resource.

## » Example Usage

```
resource "aws_opsworks_java_app_layer" "app" {
  stack_id = "${aws_opsworks_stack.main.id}"
}
```

## » Argument Reference

The following arguments are supported:

- **stack\_id** - (Required) The id of the stack the layer will belong to.
- **name** - (Optional) A human-readable name for the layer.
- **app\_server** - (Optional) Keyword for the application container to use. Defaults to "tomcat".
- **app\_server\_version** - (Optional) Version of the selected application container to use. Defaults to "7".
- **auto\_assign\_elastic\_ips** - (Optional) Whether to automatically assign an elastic IP address to the layer's instances.
- **auto\_assign\_public\_ips** - (Optional) For stacks belonging to a VPC, whether to automatically assign a public IP address to each of the layer's instances.
- **custom\_instance\_profile\_arn** - (Optional) The ARN of an IAM profile that will be used for the layer's instances.
- **custom\_security\_group\_ids** - (Optional) Ids for a set of security groups to apply to the layer's instances.

- **auto\_healing** - (Optional) Whether to enable auto-healing for the layer.
- **install\_updates\_on\_boot** - (Optional) Whether to install OS and package updates on each instance when it boots.
- **instance\_shutdown\_timeout** - (Optional) The time, in seconds, that OpsWorks will wait for Chef to complete after triggering the Shutdown event.
- **jvm\_type** - (Optional) Keyword for the type of JVM to use. Defaults to `openjdk`.
- **jvm\_options** - (Optional) Options to set for the JVM.
- **jvm\_version** - (Optional) Version of JVM to use. Defaults to `"7"`.
- **elastic\_load\_balancer** - (Optional) Name of an Elastic Load Balancer to attach to this layer
- **drain\_elb\_on\_shutdown** - (Optional) Whether to enable Elastic Load Balancing connection draining.
- **system\_packages** - (Optional) Names of a set of system packages to install on the layer's instances.
- **use\_ebs\_optimized\_instances** - (Optional) Whether to use EBS-optimized instances.
- **ebs\_volume** - (Optional) **ebs\_volume** blocks, as described below, will each create an EBS volume and connect it to the layer's instances.
- **custom\_json** - (Optional) Custom JSON attributes to apply to the layer.

The following extra optional arguments, all lists of Chef recipe names, allow custom Chef recipes to be applied to layer instances at the five different lifecycle events, if custom cookbooks are enabled on the layer's stack:

- **custom\_configure\_recipes**
- **custom\_deploy\_recipes**
- **custom\_setup\_recipes**
- **custom\_shutdown\_recipes**
- **custom\_undeploy\_recipes**

An **ebs\_volume** block supports the following arguments:

- **mount\_point** - (Required) The path to mount the EBS volume on the layer's instances.
- **size** - (Required) The size of the volume in gigabytes.
- **number\_of\_disks** - (Required) The number of disks to use for the EBS volume.
- **raid\_level** - (Required) The RAID level to use for the volume.
- **type** - (Optional) The type of volume to create. This may be **standard** (the default), **io1** or **gp2**.
- **iops** - (Optional) For PIOPS volumes, the IOPS per disk.

## » Attributes Reference

The following attributes are exported:

- `id` - The id of the layer.

## » `aws_opsworks_memcached_layer`

Provides an OpsWorks memcached layer resource.

### » Example Usage

```
resource "aws_opsworks_memcached_layer" "cache" {
  stack_id = "${aws_opsworks_stack.main.id}"
}
```

### » Argument Reference

The following arguments are supported:

- `stack_id` - (Required) The id of the stack the layer will belong to.
- `name` - (Optional) A human-readable name for the layer.
- `allocated_memory` - (Optional) Amount of memory to allocate for the cache on each instance, in megabytes. Defaults to 512MB.
- `auto_assign_elastic_ips` - (Optional) Whether to automatically assign an elastic IP address to the layer's instances.
- `auto_assign_public_ips` - (Optional) For stacks belonging to a VPC, whether to automatically assign a public IP address to each of the layer's instances.
- `custom_instance_profile_arn` - (Optional) The ARN of an IAM profile that will be used for the layer's instances.
- `custom_security_group_ids` - (Optional) Ids for a set of security groups to apply to the layer's instances.
- `auto_healing` - (Optional) Whether to enable auto-healing for the layer.
- `install_updates_on_boot` - (Optional) Whether to install OS and package updates on each instance when it boots.
- `instance_shutdown_timeout` - (Optional) The time, in seconds, that OpsWorks will wait for Chef to complete after triggering the Shutdown event.
- `elastic_load_balancer` - (Optional) Name of an Elastic Load Balancer to attach to this layer
- `drain_elb_on_shutdown` - (Optional) Whether to enable Elastic Load Balancing connection draining.
- `system_packages` - (Optional) Names of a set of system packages to install on the layer's instances.
- `use_ebs_optimized_instances` - (Optional) Whether to use EBS-optimized instances.

- `ebs_volume` - (Optional) `ebs_volume` blocks, as described below, will each create an EBS volume and connect it to the layer's instances.
- `custom_json` - (Optional) Custom JSON attributes to apply to the layer.

The following extra optional arguments, all lists of Chef recipe names, allow custom Chef recipes to be applied to layer instances at the five different lifecycle events, if custom cookbooks are enabled on the layer's stack:

- `custom_configure_recipes`
- `custom_deploy_recipes`
- `custom_setup_recipes`
- `custom_shutdown_recipes`
- `custom_undeploy_recipes`

An `ebs_volume` block supports the following arguments:

- `mount_point` - (Required) The path to mount the EBS volume on the layer's instances.
- `size` - (Required) The size of the volume in gigabytes.
- `number_of_disks` - (Required) The number of disks to use for the EBS volume.
- `raid_level` - (Required) The RAID level to use for the volume.
- `type` - (Optional) The type of volume to create. This may be `standard` (the default), `io1` or `gp2`.
- `iops` - (Optional) For PIOPS volumes, the IOPS per disk.

## » Attributes Reference

The following attributes are exported:

- `id` - The id of the layer.

## » `aws_opsworks__mysql__layer`

Provides an OpsWorks MySQL layer resource.

**Note:** All arguments including the root password will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Example Usage

```
resource "aws_opsworks_mysql_layer" "db" {
  stack_id = "${aws_opsworks_stack.main.id}"
}
```

## » Argument Reference

The following arguments are supported:

- **stack\_id** - (Required) The id of the stack the layer will belong to.
- **name** - (Optional) A human-readable name for the layer.
- **auto\_assign\_elastic\_ips** - (Optional) Whether to automatically assign an elastic IP address to the layer's instances.
- **auto\_assign\_public\_ips** - (Optional) For stacks belonging to a VPC, whether to automatically assign a public IP address to each of the layer's instances.
- **custom\_instance\_profile\_arn** - (Optional) The ARN of an IAM profile that will be used for the layer's instances.
- **custom\_security\_group\_ids** - (Optional) Ids for a set of security groups to apply to the layer's instances.
- **auto\_healing** - (Optional) Whether to enable auto-healing for the layer.
- **install\_updates\_on\_boot** - (Optional) Whether to install OS and package updates on each instance when it boots.
- **instance\_shutdown\_timeout** - (Optional) The time, in seconds, that OpsWorks will wait for Chef to complete after triggering the Shutdown event.
- **elastic\_load\_balancer** - (Optional) Name of an Elastic Load Balancer to attach to this layer
- **drain\_elb\_on\_shutdown** - (Optional) Whether to enable Elastic Load Balancing connection draining.
- **root\_password** - (Optional) Root password to use for MySQL.
- **root\_password\_on\_all\_instances** - (Optional) Whether to set the root user password to all instances in the stack so they can access the instances in this layer.
- **system\_packages** - (Optional) Names of a set of system packages to install on the layer's instances.
- **use\_ebs\_optimized\_instances** - (Optional) Whether to use EBS-optimized instances.
- **ebs\_volume** - (Optional) **ebs\_volume** blocks, as described below, will each create an EBS volume and connect it to the layer's instances.
- **custom\_json** - (Optional) Custom JSON attributes to apply to the layer.

The following extra optional arguments, all lists of Chef recipe names, allow custom Chef recipes to be applied to layer instances at the five different lifecycle events, if custom cookbooks are enabled on the layer's stack:

- **custom\_configure\_recipes**
- **custom\_deploy\_recipes**
- **custom\_setup\_recipes**
- **custom\_shutdown\_recipes**
- **custom\_undeploy\_recipes**

An `ebs_volume` block supports the following arguments:

- `mount_point` - (Required) The path to mount the EBS volume on the layer's instances.
- `size` - (Required) The size of the volume in gigabytes.
- `number_of_disks` - (Required) The number of disks to use for the EBS volume.
- `raid_level` - (Required) The RAID level to use for the volume.
- `type` - (Optional) The type of volume to create. This may be `standard` (the default), `io1` or `gp2`.
- `iops` - (Optional) For PIOPS volumes, the IOPS per disk.

## » Attributes Reference

The following attributes are exported:

- `id` - The id of the layer.

## » `aws_opsworks_nodejs_app_layer`

Provides an OpsWorks NodeJS application layer resource.

## » Example Usage

```
resource "aws_opsworks_nodejs_app_layer" "app" {
  stack_id = "${aws_opsworks_stack.main.id}"
}
```

## » Argument Reference

The following arguments are supported:

- `stack_id` - (Required) The id of the stack the layer will belong to.
- `name` - (Optional) A human-readable name for the layer.
- `auto_assign_elastic_ips` - (Optional) Whether to automatically assign an elastic IP address to the layer's instances.
- `auto_assign_public_ips` - (Optional) For stacks belonging to a VPC, whether to automatically assign a public IP address to each of the layer's instances.
- `custom_instance_profile_arn` - (Optional) The ARN of an IAM profile that will be used for the layer's instances.
- `custom_security_group_ids` - (Optional) Ids for a set of security groups to apply to the layer's instances.

- **auto\_healing** - (Optional) Whether to enable auto-healing for the layer.
- **install\_updates\_on\_boot** - (Optional) Whether to install OS and package updates on each instance when it boots.
- **instance\_shutdown\_timeout** - (Optional) The time, in seconds, that OpsWorks will wait for Chef to complete after triggering the Shutdown event.
- **elastic\_load\_balancer** - (Optional) Name of an Elastic Load Balancer to attach to this layer
- **drain\_elb\_on\_shutdown** - (Optional) Whether to enable Elastic Load Balancing connection draining.
- **nodejs\_version** - (Optional) The version of NodeJS to use. Defaults to "0.10.38".
- **system\_packages** - (Optional) Names of a set of system packages to install on the layer's instances.
- **use\_ebs\_optimized\_instances** - (Optional) Whether to use EBS-optimized instances.
- **ebs\_volume** - (Optional) **ebs\_volume** blocks, as described below, will each create an EBS volume and connect it to the layer's instances.
- **custom\_json** - (Optional) Custom JSON attributes to apply to the layer.

The following extra optional arguments, all lists of Chef recipe names, allow custom Chef recipes to be applied to layer instances at the five different lifecycle events, if custom cookbooks are enabled on the layer's stack:

- **custom\_configure\_recipes**
- **custom\_deploy\_recipes**
- **custom\_setup\_recipes**
- **custom\_shutdown\_recipes**
- **custom\_undeploy\_recipes**

An **ebs\_volume** block supports the following arguments:

- **mount\_point** - (Required) The path to mount the EBS volume on the layer's instances.
- **size** - (Required) The size of the volume in gigabytes.
- **number\_of\_disks** - (Required) The number of disks to use for the EBS volume.
- **raid\_level** - (Required) The RAID level to use for the volume.
- **type** - (Optional) The type of volume to create. This may be **standard** (the default), **io1** or **gp2**.
- **iops** - (Optional) For PIOPS volumes, the IOPS per disk.

## » Attributes Reference

The following attributes are exported:

- **id** - The id of the layer.

## » **aws\_\_opsworks\_\_permission**

Provides an OpsWorks permission resource.

### » **Example Usage**

```
resource "aws_opsworks_permission" "my_stack_permission" {
  allow_ssh  = true
  allow_sudo = true
  level      = "iam_only"
  user_arn   = "${aws_iam_user.user.arn}"
  stack_id   = "${aws_opsworks_stack.stack.id}"
}
```

### » **Argument Reference**

The following arguments are supported:

- **allow\_ssh** - (Optional) Whether the user is allowed to use SSH to communicate with the instance
- **allow\_sudo** - (Optional) Whether the user is allowed to use sudo to elevate privileges
- **user\_arn** - (Required) The user's IAM ARN to set permissions for
- **level** - (Optional) The users permission level. Must be one of **deny**, **show**, **deploy**, **manage**, **iam\_only**
- **stack\_id** - (Required) The stack to set the permissions for

### » **Attributes Reference**

The following attributes are exported:

- **id** - The computed id of the permission. Please note that this is only used internally to identify the permission. This value is not used in aws.

## » **aws\_\_opsworks\_\_php\_\_app\_\_layer**

Provides an OpsWorks PHP application layer resource.

### » **Example Usage**

```
resource "aws_opsworks_php_app_layer" "app" {
```



```
stack_id = "${aws_opsworks_stack.main.id}"
}
```

## » Argument Reference

The following arguments are supported:

- **stack\_id** - (Required) The id of the stack the layer will belong to.
- **name** - (Optional) A human-readable name for the layer.
- **auto\_assign\_elastic\_ips** - (Optional) Whether to automatically assign an elastic IP address to the layer's instances.
- **auto\_assign\_public\_ips** - (Optional) For stacks belonging to a VPC, whether to automatically assign a public IP address to each of the layer's instances.
- **custom\_instance\_profile\_arn** - (Optional) The ARN of an IAM profile that will be used for the layer's instances.
- **custom\_security\_group\_ids** - (Optional) Ids for a set of security groups to apply to the layer's instances.
- **auto\_healing** - (Optional) Whether to enable auto-healing for the layer.
- **install\_updates\_on\_boot** - (Optional) Whether to install OS and package updates on each instance when it boots.
- **instance\_shutdown\_timeout** - (Optional) The time, in seconds, that OpsWorks will wait for Chef to complete after triggering the Shutdown event.
- **elastic\_load\_balancer** - (Optional) Name of an Elastic Load Balancer to attach to this layer
- **drain\_elb\_on\_shutdown** - (Optional) Whether to enable Elastic Load Balancing connection draining.
- **system\_packages** - (Optional) Names of a set of system packages to install on the layer's instances.
- **use\_ebs\_optimized\_instances** - (Optional) Whether to use EBS-optimized instances.
- **ebs\_volume** - (Optional) **ebs\_volume** blocks, as described below, will each create an EBS volume and connect it to the layer's instances.
- **custom\_json** - (Optional) Custom JSON attributes to apply to the layer.

The following extra optional arguments, all lists of Chef recipe names, allow custom Chef recipes to be applied to layer instances at the five different lifecycle events, if custom cookbooks are enabled on the layer's stack:

- **custom\_configure\_recipes**
- **custom\_deploy\_recipes**
- **custom\_setup\_recipes**
- **custom\_shutdown\_recipes**
- **custom\_undeploy\_recipes**

An **ebs\_volume** block supports the following arguments:

- **mount\_point** - (Required) The path to mount the EBS volume on the layer's instances.
- **size** - (Required) The size of the volume in gigabytes.
- **number\_of\_disks** - (Required) The number of disks to use for the EBS volume.
- **raid\_level** - (Required) The RAID level to use for the volume.
- **type** - (Optional) The type of volume to create. This may be **standard** (the default), **io1** or **gp2**.
- **iops** - (Optional) For PIOPS volumes, the IOPS per disk.

## » Attributes Reference

The following attributes are exported:

- **id** - The id of the layer.

## » `aws_opsworks_rails_app_layer`

Provides an OpsWorks Ruby on Rails application layer resource.

## » Example Usage

```
resource "aws_opsworks_rails_app_layer" "app" {
  stack_id = "${aws_opsworks_stack.main.id}"
}
```

## » Argument Reference

The following arguments are supported:

- **stack\_id** - (Required) The id of the stack the layer will belong to.
- **name** - (Optional) A human-readable name for the layer.
- **app\_server** - (Optional) Keyword for the app server to use. Defaults to "apache\_passenger".
- **auto\_assign\_elastic\_ips** - (Optional) Whether to automatically assign an elastic IP address to the layer's instances.
- **auto\_assign\_public\_ips** - (Optional) For stacks belonging to a VPC, whether to automatically assign a public IP address to each of the layer's instances.
- **bundler\_version** - (Optional) When OpsWorks is managing Bundler, which version to use. Defaults to "1.5.3".

- `custom_instance_profile_arn` - (Optional) The ARN of an IAM profile that will be used for the layer's instances.
- `custom_security_group_ids` - (Optional) Ids for a set of security groups to apply to the layer's instances.
- `auto_healing` - (Optional) Whether to enable auto-healing for the layer.
- `install_updates_on_boot` - (Optional) Whether to install OS and package updates on each instance when it boots.
- `instance_shutdown_timeout` - (Optional) The time, in seconds, that OpsWorks will wait for Chef to complete after triggering the Shutdown event.
- `elastic_load_balancer` - (Optional) Name of an Elastic Load Balancer to attach to this layer
- `drain_elb_on_shutdown` - (Optional) Whether to enable Elastic Load Balancing connection draining.
- `manage_bundler` - (Optional) Whether OpsWorks should manage bundler. On by default.
- `passenger_version` - (Optional) The version of Passenger to use. Defaults to "4.0.46".
- `ruby_version` - (Optional) The version of Ruby to use. Defaults to "2.0.0".
- `rubygems_version` - (Optional) The version of RubyGems to use. Defaults to "2.2.2".
- `system_packages` - (Optional) Names of a set of system packages to install on the layer's instances.
- `use_ebs_optimized_instances` - (Optional) Whether to use EBS-optimized instances.
- `ebs_volume` - (Optional) `ebs_volume` blocks, as described below, will each create an EBS volume and connect it to the layer's instances.
- `custom_json` - (Optional) Custom JSON attributes to apply to the layer.

The following extra optional arguments, all lists of Chef recipe names, allow custom Chef recipes to be applied to layer instances at the five different lifecycle events, if custom cookbooks are enabled on the layer's stack:

- `custom_configure_recipes`
- `custom_deploy_recipes`
- `custom_setup_recipes`
- `custom_shutdown_recipes`
- `custom_undeploy_recipes`

An `ebs_volume` block supports the following arguments:

- `mount_point` - (Required) The path to mount the EBS volume on the layer's instances.
- `size` - (Required) The size of the volume in gigabytes.
- `number_of_disks` - (Required) The number of disks to use for the EBS volume.
- `raid_level` - (Required) The RAID level to use for the volume.

- **type** - (Optional) The type of volume to create. This may be **standard** (the default), **io1** or **gp2**.
- **iops** - (Optional) For PIOPS volumes, the IOPS per disk.

## » Attributes Reference

The following attributes are exported:

- **id** - The id of the layer.

## » `aws__opsworks__rds__db__instance`

Provides an OpsWorks RDS DB Instance resource.

**Note:** All arguments including the username and password will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Example Usage

```
resource "aws_opsworks_rds_db_instance" "my_instance" {
  stack_id          = "${aws_opsworks_stack.my_stack.id}"
  rds_db_instance_arn = "${aws_db_instance.my_instance.arn}"
  db_user           = "someUser"
  db_password       = "somePass"
}
```

## » Argument Reference

The following arguments are supported:

- **stack\_id** - (Required) The stack to register a db instance for. Changing this will force a new resource.
- **rds\_db\_instance\_arn** - (Required) The db instance to register for this stack. Changing this will force a new resource.
- **db\_user** - (Required) A db username
- **db\_password** - (Required) A db password

## » Attributes Reference

The following attributes are exported:

- **id** - The computed id. Please note that this is only used internally to identify the stack <-> instance relation. This value is not used in aws.

## » **aws\_\_opsworks\_\_stack**

Provides an OpsWorks stack resource.

### » **Example Usage**

```
resource "aws_opsworks_stack" "main" {
  name                = "awesome-stack"
  region              = "us-west-1"
  service_role_arn    = "${aws_iam_role.opsworks.arn}"
  default_instance_profile_arn = "${aws_iam_instance_profile.opsworks.arn}"

  tags {
    Name = "foobar-terraform-stack"
  }

  custom_json = <<EOT
{
  "foobar": {
    "version": "1.0.0"
  }
}
EOT
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) The name of the stack.
- **region** - (Required) The name of the region where the stack will exist.
- **service\_role\_arn** - (Required) The ARN of an IAM role that the OpsWorks service will act as.
- **default\_instance\_profile\_arn** - (Required) The ARN of an IAM Instance Profile that created instances will have by default.
- **agent\_version** - (Optional) If set to "LATEST", OpsWorks will automatically install the latest version.
- **berkshelf\_version** - (Optional) If **manage\_berkshelf** is enabled, the version of Berkshelf to use.
- **color** - (Optional) Color to paint next to the stack's resources in the OpsWorks console.
- **default\_availability\_zone** - (Optional) Name of the availability zone where instances will be created by default. This is required unless you set **vpc\_id**.

- `configuration_manager_name` - (Optional) Name of the configuration manager to use. Defaults to "Chef".
- `configuration_manager_version` - (Optional) Version of the configuration manager to use. Defaults to "11.4".
- `custom_cookbooks_source` - (Optional) When `use_custom_cookbooks` is set, provide this sub-object as described below.
- `custom_json` - (Optional) User defined JSON passed to "Chef". Use a "here doc" for multiline JSON.
- `default_os` - (Optional) Name of OS that will be installed on instances by default.
- `default_root_device_type` - (Optional) Name of the type of root device instances will have by default.
- `default_ssh_key_name` - (Optional) Name of the SSH keypair that instances will have by default.
- `default_subnet_id` - (Optional) Id of the subnet in which instances will be created by default. Mandatory if `vpc_id` is set, and forbidden if it isn't.
- `hostname_theme` - (Optional) Keyword representing the naming scheme that will be used for instance hostnames within this stack.
- `manage_berkshelf` - (Optional) Boolean value controlling whether Opsworks will run Berkshelf for this stack.
- `tags` - (Optional) A mapping of tags to assign to the resource.
- `use_custom_cookbooks` - (Optional) Boolean value controlling whether the custom cookbook settings are enabled.
- `use_opsworks_security_groups` - (Optional) Boolean value controlling whether the standard OpsWorks security groups apply to created instances.
- `vpc_id` - (Optional) The id of the VPC that this stack belongs to.
- `custom_json` - (Optional) Custom JSON attributes to apply to the entire stack.

The `custom_cookbooks_source` block supports the following arguments:

- `type` - (Required) The type of source to use. For example, "archive".
- `url` - (Required) The URL where the cookbooks resource can be found.
- `username` - (Optional) Username to use when authenticating to the source.
- `password` - (Optional) Password to use when authenticating to the source.
- `ssh_key` - (Optional) SSH key to use when authenticating to the source.
- `revision` - (Optional) For sources that are version-aware, the revision to use.

## » Attributes Reference

The following attributes are exported:

- `id` - The id of the stack.

## » Import

OpsWorks stacks can be imported using the `id`, e.g.

```
$ terraform import aws_opsworks_stack.bar 00000000-0000-0000-0000-000000000000
```

## » `aws_opsworks_static_web_layer`

Provides an OpsWorks static web server layer resource.

## » Example Usage

```
resource "aws_opsworks_static_web_layer" "web" {  
  stack_id = "${aws_opsworks_stack.main.id}"  
}
```

## » Argument Reference

The following arguments are supported:

- `stack_id` - (Required) The id of the stack the layer will belong to.
- `name` - (Optional) A human-readable name for the layer.
- `auto_assign_elastic_ips` - (Optional) Whether to automatically assign an elastic IP address to the layer's instances.
- `auto_assign_public_ips` - (Optional) For stacks belonging to a VPC, whether to automatically assign a public IP address to each of the layer's instances.
- `custom_instance_profile_arn` - (Optional) The ARN of an IAM profile that will be used for the layer's instances.
- `custom_security_group_ids` - (Optional) Ids for a set of security groups to apply to the layer's instances.
- `auto_healing` - (Optional) Whether to enable auto-healing for the layer.
- `install_updates_on_boot` - (Optional) Whether to install OS and package updates on each instance when it boots.
- `instance_shutdown_timeout` - (Optional) The time, in seconds, that OpsWorks will wait for Chef to complete after triggering the Shutdown event.
- `elastic_load_balancer` - (Optional) Name of an Elastic Load Balancer to attach to this layer
- `drain_elb_on_shutdown` - (Optional) Whether to enable Elastic Load Balancing connection draining.
- `system_packages` - (Optional) Names of a set of system packages to install on the layer's instances.

- `use_ebs_optimized_instances` - (Optional) Whether to use EBS-optimized instances.
- `ebs_volume` - (Optional) `ebs_volume` blocks, as described below, will each create an EBS volume and connect it to the layer's instances.

The following extra optional arguments, all lists of Chef recipe names, allow custom Chef recipes to be applied to layer instances at the five different lifecycle events, if custom cookbooks are enabled on the layer's stack:

- `custom_configure_recipes`
- `custom_deploy_recipes`
- `custom_setup_recipes`
- `custom_shutdown_recipes`
- `custom_undeploy_recipes`

An `ebs_volume` block supports the following arguments:

- `mount_point` - (Required) The path to mount the EBS volume on the layer's instances.
- `size` - (Required) The size of the volume in gigabytes.
- `number_of_disks` - (Required) The number of disks to use for the EBS volume.
- `raid_level` - (Required) The RAID level to use for the volume.
- `type` - (Optional) The type of volume to create. This may be `standard` (the default), `io1` or `gp2`.
- `iops` - (Optional) For PIOPS volumes, the IOPS per disk.

## » Attributes Reference

The following attributes are exported:

- `id` - The id of the layer.

## » `aws__opsworks__user__profile`

Provides an OpsWorks User Profile resource.

## » Example Usage

```
resource "aws_opsworks_user_profile" "my_profile" {
  user_arn      = "${aws_iam_user.user.arn}"
  ssh_username = "my_user"
}
```



## » Argument Reference

The following arguments are supported:

- **user\_arn** - (Required) The user's IAM ARN
- **allow\_self\_management** - (Optional) Whether users can specify their own SSH public key through the My Settings page
- **ssh\_username** - (Required) The ssh username, with which this user wants to log in
- **ssh\_public\_key** - (Optional) The user's public key

## » Attributes Reference

The following attributes are exported:

- **id** - Same value as **user\_arn**

## » `aws_organizations_account`

Provides a resource to create a member account in the current organization.

**Note:** Account management must be done from the organization's master account.

**WARNING:** Deleting this Terraform resource will only remove an AWS account from an organization. Terraform will not close the account. The member account must be prepared to be a standalone account beforehand. See the AWS Organizations documentation for more information.

## » Example Usage:

```
resource "aws_organizations_account" "account" {  
  name = "my_new_account"  
  email = "john@doe.org"  
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A friendly name for the member account.
- **email** - (Required) The email address of the owner to assign to the new member account. This email address must not already be associated with another AWS account.

- **iam\_user\_access\_to\_billing** - (Optional) If set to **ALLOW**, the new account enables IAM users to access account billing information if they have the required permissions. If set to **DENY**, then only the root user of the new account can access account billing information.
- **role\_name** - (Optional) The name of an IAM role that Organizations automatically preconfigures in the new member account. This role trusts the master account, allowing users in the master account to assume the role, as permitted by the master account administrator. The role has administrator permissions in the new member account.

## » Attributes Reference

The following additional attributes are exported:

- **arn** - The ARN for this account.

## » Import

The AWS member account can be imported by using the **account\_id**, e.g.

```
$ terraform import aws_organizations_account.my_org 111111111111
```

## » **aws\_organizations\_organization**

Provides a resource to create an organization.

## » Example Usage:

```
resource "aws_organizations_organization" "org" {
  feature_set = "ALL"
}
```

## » Argument Reference

The following arguments are supported:

- **feature\_set** - (Optional) Specify "ALL" (default) or "CONSOLIDATED\_BILLING".

## » Attributes Reference

The following additional attributes are exported:

- `arn` - ARN of the organization
- `id` - Identifier of the organization
- `master_account_arn` - ARN of the master account
- `master_account_email` - Email address of the master account
- `master_account_id` - Identifier of the master account

## » Import

The AWS organization can be imported by using the `id`, e.g.

```
$ terraform import aws_organizations_organization.my_org o-1234567
```

## » `aws_organizations_policy`

Provides a resource to manage an AWS Organizations policy.

## » Example Usage

```
resource "aws_organizations_policy" "example" {
  name      = "example"

  content = <<CONTENT
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": "*",
    "Resource": "*"
  }
}
CONTENT
}
```

## » Argument Reference

The following arguments are supported:

- **content** - (Required) The policy content to add to the new policy. For example, if you create a service control policy (SCP), this string must be JSON text that specifies the permissions that admins in attached accounts can delegate to their users, groups, and roles. For more information about the SCP syntax, see the [Service Control Policy Syntax documentation]((https://docs.aws.amazon.com/organizations/latest/userguide/orgs\_reference\_scp-syntax.html)).
- **name** - (Required) The friendly name to assign to the policy.
- **description** - (Optional) A description to assign to the policy.
- **type** - (Optional) The type of policy to create. Currently, the only valid value is `SERVICE_CONTROL_POLICY` (SCP).

## » Attribute Reference

- **id** - The unique identifier (ID) of the policy.
- **arn** - Amazon Resource Name (ARN) of the policy.

## » Import

`aws_organizations_policy` can be imported by using the policy ID, e.g.

```
$ terraform import aws_organizations_policy.example p-12345678
```

## » `aws_organizations_policy_attachment`

Provides a resource to attach an AWS Organizations policy to an organization account, root, or unit.

## » Example Usage

### » Organization Account

```
resource "aws_organizations_policy_attachment" "account" {
  policy_id = "${aws_organizations_policy.example.id}"
  target_id = "123456789012"
}
```

### » Organization Root

```
resource "aws_organizations_policy_attachment" "root" {
  policy_id = "${aws_organizations_policy.example.id}"
}
```

```

    target_id = "r-12345678"
}

```

## » Organization Unit

```

resource "aws_organizations_policy_attachment" "unit" {
  policy_id = "${aws_organizations_policy.example.id}"
  target_id = "ou-12345678"
}

```

## » Argument Reference

The following arguments are supported:

- `policy_id` - (Required) The unique identifier (ID) of the policy that you want to attach to the target.
- `target_id` - (Required) The unique identifier (ID) of the root, organizational unit, or account number that you want to attach the policy to.

## » Import

`aws_organizations_policy_attachment` can be imported by using the target ID and policy ID, e.g. with an account target

```
$ terraform import aws_organization_policy_attachment.account 123456789012:p-12345678
```

## » `aws_db_event_subscription`

Provides a DB event subscription resource.

## » Example Usage

```

resource "aws_db_instance" "default" {
  allocated_storage    = 10
  engine               = "mysql"
  engine_version       = "5.6.17"
  instance_class       = "db.t2.micro"
  name                 = "mydb"
  username             = "foo"
  password             = "bar"
  db_subnet_group_name = "my_database_subnet_group"
  parameter_group_name = "default.mysql5.6"
}

```

```

}

resource "aws_sns_topic" "default" {
  name = "rds-events"
}

resource "aws_db_event_subscription" "default" {
  name          = "rds-event-sub"
  sns_topic     = "${aws_sns_topic.default.arn}"

  source_type = "db-instance"
  source_ids  = ["${aws_db_instance.default.id}"]

  event_categories = [
    "availability",
    "deletion",
    "failover",
    "failure",
    "low storage",
    "maintenance",
    "notification",
    "read replica",
    "recovery",
    "restoration",
  ]
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the DB event subscription.
- **sns\_topic** - (Required) The SNS topic to send events to.
- **source\_ids** - (Optional) A list of identifiers of the event sources for which events will be returned. If not specified, then all sources are included in the response. If specified, a **source\_type** must also be specified.
- **source\_type** - (Optional) The type of source that will be generating the events.
- **event\_categories** - (Optional) A list of event categories for a Source-Type that you want to subscribe to. See [http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide//USER\\_Events.html](http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide//USER_Events.html)
- **enabled** - (Optional) A boolean flag to enable/disable the subscription. Defaults to true.
- **tags** - (Optional) A mapping of tags to assign to the resource.

## » Attributes

The following additional attributes are provided:

- **id** - The name of the RDS event notification subscription
- **arn** - The Amazon Resource Name of the RDS event notification subscription
- **customer\_aws\_id** - The AWS customer account associated with the RDS event notification subscription

## » Timeouts

`aws_db_event_subscription` provides the following Timeouts configuration options:

- **create** - (Default 40m) How long to wait for a RDS event notification subscription to be ready.
- **delete** - (Default 40m) How long to wait for a RDS event notification subscription to be deleted.
- **update** - (Default 40m) How long to wait for a RDS event notification subscription to be updated.

## » Import

DB Event Subscriptions can be imported using the **name**, e.g.

```
$ terraform import aws_db_event_subscription.default rds-event-sub
```

## » `aws_db_instance`

Provides an RDS instance resource. A DB instance is an isolated database environment in the cloud. A DB instance can contain multiple user-created databases.

Changes to a DB instance can occur when you manually change a parameter, such as `allocated_storage`, and are reflected in the next maintenance window. Because of this, Terraform may report a difference in its planning phase because a modification has not yet taken place. You can use the `apply_immediately` flag to instruct the service to apply the change immediately (see documentation below).

When upgrading the major version of an engine, `allow_major_version_upgrade` must be set to `true`.

**Note:** using `apply_immediately` can result in a brief downtime as the server reboots. See the AWS Docs on RDS Maintenance for more information.

**Note:** All arguments including the username and password will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » RDS Instance Class Types

Amazon RDS supports three types of instance classes: Standard, Memory Optimized, and Burstable Performance. For more information please read the AWS RDS documentation about DB Instance Class Types

## » Example Usage

```
resource "aws_db_instance" "default" {
  allocated_storage    = 10
  storage_type         = "gp2"
  engine               = "mysql"
  engine_version       = "5.7"
  instance_class       = "db.t2.micro"
  name                 = "mydb"
  username              = "foo"
  password              = "foobaz"
  parameter_group_name = "default.mysql5.7"
}
```

## » Argument Reference

For more detailed documentation about each argument, refer to the AWS official documentation.

The following arguments are supported:

- `allocated_storage` - (Required unless a `snapshot_identifier` or `replicate_source_db` is provided) The allocated storage in gigabytes.
- `allow_major_version_upgrade` - (Optional) Indicates that major version upgrades are allowed. Changing this parameter does not result in an outage and the change is asynchronously applied as soon as possible.
- `apply_immediately` - (Optional) Specifies whether any database modifications are applied immediately, or during the next maintenance window. Default is `false`. See Amazon RDS Documentation for more information.
- `auto_minor_version_upgrade` - (Optional) Indicates that minor engine upgrades will be applied automatically to the DB instance during the maintenance window. Defaults to `true`.



- **availability\_zone** - (Optional) The AZ for the RDS instance.
- **backup\_retention\_period** - (Optional) The days to retain backups for. Must be 1 or greater to be a source for a Read Replica.
- **backup\_window** - (Optional) The daily time range (in UTC) during which automated backups are created if they are enabled. Example: "09:46-10:16". Must not overlap with **maintenance\_window**.
- **character\_set\_name** - (Optional) The character set name to use for DB encoding in Oracle instances. This can't be changed. See Oracle Character Sets Supported in Amazon RDS for more information.
- **copy\_tags\_to\_snapshot** - (Optional, boolean) On delete, copy all Instance tags to the final snapshot (if **final\_snapshot\_identifier** is specified). Default is **false**.
- **db\_subnet\_group\_name** - (Optional) Name of DB subnet group. DB instance will be created in the VPC associated with the DB subnet group. If unspecified, will be created in the **default** VPC, or in EC2 Classic, if available. When working with read replicas, it needs to be specified only if the source database specifies an instance in another AWS Region. See DBSubnetGroupName in API action CreateDBInstanceReadReplica for additional read replica constraints.
- **enabled\_cloudwatch\_logs\_exports** - (Optional) Name list of enable log type for exporting to cloudwatch logs. If omitted, any logs will not be exported to cloudwatch logs. Either of the following is supported: **audit**, **error**, **general**, **slowquery**.
- **engine** - (Required unless a **snapshot\_identifier** or **replicate\_source\_db** is provided) The database engine to use. For supported values, see the Engine parameter in API action CreateDBInstance.
- **engine\_version** - (Optional) The engine version to use. If **auto\_minor\_version\_upgrade** is enabled, you can provide a prefix of the version such as 5.7 (for 5.7.10) and this attribute will ignore differences in the patch version automatically (e.g. 5.7.17).
- **final\_snapshot\_identifier** - (Optional) The name of your final DB snapshot when this DB instance is deleted. If omitted, no final snapshot will be made.
- **iam\_database\_authentication\_enabled** - (Optional) Specifies whether or mappings of AWS Identity and Access Management (IAM) accounts to database accounts is enabled.
- **identifier** - (Optional, Forces new resource) The name of the RDS instance, if omitted, Terraform will assign a random, unique identifier.
- **identifier\_prefix** - (Optional, Forces new resource) Creates a unique identifier beginning with the specified prefix. Conflicts with **identifier**.
- **instance\_class** - (Required) The instance type of the RDS instance.
- **iops** - (Optional) The amount of provisioned IOPS. Setting this implies a **storage\_type** of "io1".
- **kms\_key\_id** - (Optional) The ARN for the KMS encryption key. If creating an encrypted replica, set this to the destination KMS ARN.
- **license\_model** - (Optional, but required for some DB engines, i.e. Oracle

SE1) License model information for this DB instance.

- **maintenance\_window** - (Optional) The window to perform maintenance in. Syntax: "ddd:hh24:mi-ddd:hh24:mi". Eg: "Mon:00:00-Mon:03:00". See RDS Maintenance Window docs for more information.
- **monitoring\_interval** - (Optional) The interval, in seconds, between points when Enhanced Monitoring metrics are collected for the DB instance. To disable collecting Enhanced Monitoring metrics, specify 0. The default is 0. Valid Values: 0, 1, 5, 10, 15, 30, 60.
- **monitoring\_role\_arn** - (Optional) The ARN for the IAM role that permits RDS to send enhanced monitoring metrics to CloudWatch Logs. You can find more information on the AWS Documentation what IAM permissions are needed to allow Enhanced Monitoring for RDS Instances.
- **multi\_az** - (Optional) Specifies if the RDS instance is multi-AZ
- **name** - (Optional) The name of the database to create when the DB instance is created. If this parameter is not specified, no database is created in the DB instance. Note that this does not apply for Oracle or SQL Server engines. See the AWS documentation for more details on what applies for those engines.
- **option\_group\_name** - (Optional) Name of the DB option group to associate.
- **parameter\_group\_name** - (Optional) Name of the DB parameter group to associate.
- **password** - (Required unless a **snapshot\_identifier** or **replicate\_source\_db** is provided) Password for the master DB user. Note that this may show up in logs, and it will be stored in the state file.
- **port** - (Optional) The port on which the DB accepts connections.
- **publicly\_accessible** - (Optional) Bool to control if instance is publicly accessible. Default is **false**.
- **replicate\_source\_db** - (Optional) Specifies that this resource is a Replicate database, and to use this value as the source database. This correlates to the **identifier** of another Amazon RDS Database to replicate. Note that if you are creating a cross-region replica of an encrypted database you will also need to specify a **kms\_key\_id**. See DB Instance Replication and Working with PostgreSQL and MySQL Read Replicas for more information on using Replication.
- **security\_group\_names** - (Optional/Deprecated) List of DB Security Groups to associate. Only used for DB Instances on the *EC2-Classical* Platform.
- **skip\_final\_snapshot** - (Optional) Determines whether a final DB snapshot is created before the DB instance is deleted. If true is specified, no DBSnapshot is created. If false is specified, a DB snapshot is created before the DB instance is deleted, using the value from **final\_snapshot\_identifier**. Default is **false**.
- **snapshot\_identifier** - (Optional) Specifies whether or not to create this database from a snapshot. This correlates to the snapshot ID you'd find in the RDS console, e.g: rds:production-2015-06-26-06-05.

- **storage\_encrypted** - (Optional) Specifies whether the DB instance is encrypted. Note that if you are creating a cross-region read replica this field is ignored and you should instead declare **kms\_key\_id** with a valid ARN. The default is **false** if not specified.
- **storage\_type** - (Optional) One of "standard" (magnetic), "gp2" (general purpose SSD), or "io1" (provisioned IOPS SSD). The default is "io1" if **iops** is specified, "standard" if not. Note that this behaviour is different from the AWS web console, where the default is "gp2".
- **tags** - (Optional) A mapping of tags to assign to the resource.
- **timezone** - (Optional) Time zone of the DB instance. **timezone** is currently only supported by Microsoft SQL Server. The **timezone** can only be set on creation. See MSSQL User Guide for more information.
- **username** - (Required unless a **snapshot\_identifier** or **replicate\_source\_db** is provided) Username for the master DB user.
- **vpc\_security\_group\_ids** - (Optional) List of VPC security groups to associate.
- **s3\_import** - (Optional) Restore from a Percona Xtrabackup in S3. See Importing Data into an Amazon RDS MySQL DB Instance

**NOTE:** Removing the **replicate\_source\_db** attribute from an existing RDS Replicate database managed by Terraform will promote the database to a fully standalone database.

## » S3 Import Options

Full details on the core parameters and impacts are in the API Docs: [RestoreDBInstanceFromS3](#). Sample

```
resource "aws_db_instance" "db" {
  s3_import {
    source_engine = "mysql"
    source_engine_version = "5.6"
    bucket_name = "mybucket"
    bucket_prefix = "backups"
    ingestion_role = "arn:aws:iam::1234567890:role/role-xtrabackup-rds-restore"
  }
}
```

- **bucket\_name** - (Required) The bucket name where your backup is stored
- **bucket\_prefix** - (Optional) Can be blank, but is the path to your backup
- **ingestion\_role** - (Required) Role applied to load the data.
- **source\_engine** - (Required, as of Feb 2018 only 'mysql' supported) Source engine for the backup
- **source\_engine\_version** - (Required, as of Feb 2018 only '5.6' supported) Version of the source engine used to make the backup

This will not recreate the resource if the S3 object changes in some way. It's only used to initialize the database

## » Timeouts

`aws_db_instance` provides the following Timeouts configuration options:

- `create` - (Default 40 minutes) Used for Creating Instances, Replicas, and restoring from Snapshots.
- `update` - (Default 80 minutes) Used for Database modifications.
- `delete` - (Default 40 minutes) Used for destroying databases. This includes the time required to take snapshots.

## » Attributes Reference

The following attributes are exported:

- `address` - The address of the RDS instance.
- `arn` - The ARN of the RDS instance.
- `allocated_storage` - The amount of allocated storage.
- `availability_zone` - The availability zone of the instance.
- `backup_retention_period` - The backup retention period.
- `backup_window` - The backup window.
- `ca_cert_identifier` - Specifies the identifier of the CA certificate for the DB instance.
- `endpoint` - The connection endpoint.
- `engine` - The database engine.
- `engine_version` - The database engine version.
- `hosted_zone_id` - The canonical hosted zone ID of the DB instance (to be used in a Route 53 Alias record).
- `id` - The RDS instance ID.
- `instance_class` - The RDS instance class.
- `maintenance_window` - The instance maintenance window.
- `multi_az` - If the RDS instance is multi AZ enabled.
- `name` - The database name.
- `port` - The database port.
- `resource_id` - The RDS Resource ID of this instance.
- `status` - The RDS instance status.
- `storage_encrypted` - Specifies whether the DB instance is encrypted.
- `username` - The master username for the database.

On Oracle instances the following is exported additionally:

- `character_set_name` - The character set used on Oracle instances.

## » Import

DB Instances can be imported using the `identifier`, e.g.

```
$ terraform import aws_db_instance.default mydb-rds-instance
```

## » aws\_db\_option\_group

Provides an RDS DB option group resource.

## » Example Usage

```
resource "aws_db_option_group" "bar" {
  name                     = "option-group-test-terraform"
  option_group_description = "Terraform Option Group"
  engine_name              = "sqlserver-ee"
  major_engine_version     = "11.00"

  option {
    option_name = "Timezone"

    option_settings {
      name = "TIME_ZONE"
      value = "UTC"
    }
  }

  option {
    option_name = "TDE"
  }
}
```

**Note:** Any modifications to the `db_option_group` are set to happen immediately as we default to applying immediately.

## » Argument Reference

The following arguments are supported:

- `name` - (Optional, Forces new resource) The name of the option group. If omitted, Terraform will assign a random, unique name. Must be lowercase, to match as it is stored in AWS.

- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**. Must be lowercase, to match as it is stored in AWS.
- **option\_group\_description** - (Optional) The description of the option group. Defaults to "Managed by Terraform".
- **engine\_name** - (Required) Specifies the name of the engine that this option group should be associated with.
- **major\_engine\_version** - (Required) Specifies the major version of the engine that this option group should be associated with.
- **option** - (Optional) A list of Options to apply.
- **tags** - (Optional) A mapping of tags to assign to the resource.

Option blocks support the following:

- **option\_name** - (Required) The Name of the Option (e.g. MEMCACHED).
- **option\_settings** - (Optional) A list of option settings to apply.
- **port** - (Optional) The Port number when connecting to the Option (e.g. 11211).
- **version** - (Optional) The version of the option (e.g. 13.1.0.0).
- **db\_security\_group\_memberships** - (Optional) A list of DB Security Groups for which the option is enabled.
- **vpc\_security\_group\_memberships** - (Optional) A list of VPC Security Groups for which the option is enabled.

Option Settings blocks support the following:

- **name** - (Optional) The Name of the setting.
- **value** - (Optional) The Value of the setting.

## » Attributes Reference

The following attributes are exported:

- **id** - The db option group name.
- **arn** - The ARN of the db option group.

## » Timeouts

`aws_db_option_group` provides the following Timeouts configuration options:

- **delete** - (Default 15 minutes)

## » Import

DB Option groups can be imported using the **name**, e.g.

```
$ terraform import aws_db_option_group.bar mysql-option-group
```

## » `aws_db_parameter_group`

Provides an RDS DB parameter group resource.

### » Example Usage

```
resource "aws_db_parameter_group" "default" {
  name     = "rds-pg"
  family   = "mysql5.6"

  parameter {
    name = "character_set_server"
    value = "utf8"
  }

  parameter {
    name = "character_set_client"
    value = "utf8"
  }
}
```

### » Argument Reference

The following arguments are supported:

- **name** - (Optional, Forces new resource) The name of the DB parameter group. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **family** - (Required) The family of the DB parameter group.
- **description** - (Optional) The description of the DB parameter group. Defaults to "Managed by Terraform".
- **parameter** - (Optional) A list of DB parameters to apply. Note that parameters may differ from a family to another. Full list of all parameters can be discovered via `aws rds describe-db-parameters` after initial creation of the group.
- **tags** - (Optional) A mapping of tags to assign to the resource.

Parameter blocks support the following:

- **name** - (Required) The name of the DB parameter.
- **value** - (Required) The value of the DB parameter.

- `apply_method` - (Optional) "immediate" (default), or "pending-reboot". Some engines can't apply some parameters without a reboot, and you will need to specify "pending-reboot" here.

## » Attributes Reference

The following attributes are exported:

- `id` - The db parameter group name.
- `arn` - The ARN of the db parameter group.

## » Import

DB Parameter groups can be imported using the `name`, e.g.

```
$ terraform import aws_db_parameter_group.rds_pg rds-pg
```

## » `aws_db_security_group`

Provides an RDS security group resource. This is only for DB instances in the EC2-Classic Platform. For instances inside a VPC, use the `aws_db_instance.vpc_security_group_ids` attribute instead.

## » Example Usage

```
resource "aws_db_security_group" "default" {
  name = "rds_sg"

  ingress {
    cidr = "10.0.0.0/24"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the DB security group.
- `description` - (Optional) The description of the DB security group. Defaults to "Managed by Terraform".
- `ingress` - (Required) A list of ingress rules.
- `tags` - (Optional) A mapping of tags to assign to the resource.



Ingress blocks support the following:

- `cidr` - The CIDR block to accept
- `security_group_name` - The name of the security group to authorize
- `security_group_id` - The ID of the security group to authorize
- `security_group_owner_id` - The owner Id of the security group provided by `security_group_name`.

## » Attributes Reference

The following attributes are exported:

- `id` - The db security group ID.
- `arn` - The arn of the DB security group.

## » Import

DB Security groups can be imported using the `name`, e.g.

```
$ terraform import aws_db_security_group.default aws_rds_sg-1
```

## » `aws_db_snapshot`

Creates a Snapshot of an DB Instance.

## » Example Usage

```
resource "aws_db_instance" "bar" {
  allocated_storage = 10
  engine            = "MySQL"
  engine_version   = "5.6.21"
  instance_class    = "db.t2.micro"
  name              = "baz"
  password          = "barbarbarbar"
  username          = "foo"

  maintenance_window = "Fri:09:00-Fri:09:30"
  backup_retention_period = 0
  parameter_group_name = "default.mysql5.6"
}

resource "aws_db_snapshot" "test" {
  db_instance_identifier = "${aws_db_instance.bar.id}"
}
```

```

    db_snapshot_identifier = "testsnapshot1234"
}

```

## » Argument Reference

The following arguments are supported:

- **db\_instance\_identifier** - (Required) The DB Instance Identifier from which to take the snapshot.
- **db\_snapshot\_identifier** - (Required) The Identifier for the snapshot.

## » Attributes Reference

The following attributes are exported:

- **allocated\_storage** - Specifies the allocated storage size in gigabytes (GB).
- **availability\_zone** - Specifies the name of the Availability Zone the DB instance was located in at the time of the DB snapshot.
- **db\_snapshot\_arn** - The Amazon Resource Name (ARN) for the DB snapshot.
- **encrypted** - Specifies whether the DB snapshot is encrypted.
- **engine** - Specifies the name of the database engine.
- **engine\_version** - Specifies the version of the database engine.
- **iops** - Specifies the Provisioned IOPS (I/O operations per second) value of the DB instance at the time of the snapshot.
- **kms\_key\_id** - The ARN for the KMS encryption key.
- **license\_model** - License model information for the restored DB instance.
- **option\_group\_name** - Provides the option group name for the DB snapshot.
- **source\_db\_snapshot\_identifier** - The DB snapshot Arn that the DB snapshot was copied from. It only has value in case of cross customer or cross region copy.
- **source\_region** - The region that the DB snapshot was created in or copied from.
- **status** - Specifies the status of this DB snapshot.
- **storage\_type** - Specifies the storage type associated with DB snapshot.
- **vpc\_id** - Specifies the storage type associated with DB snapshot.

## » aws\_db\_subnet\_group

Provides an RDS DB subnet group resource.

## » Example Usage

```
resource "aws_db_subnet_group" "default" {
  name          = "main"
  subnet_ids    = ["${aws_subnet.frontend.id}", "${aws_subnet.backend.id}"]

  tags {
    Name = "My DB subnet group"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional, Forces new resource) The name of the DB subnet group. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **description** - (Optional) The description of the DB subnet group. Defaults to "Managed by Terraform".
- **subnet\_ids** - (Required) A list of VPC subnet IDs.
- **tags** - (Optional) A mapping of tags to assign to the resource.

## » Attributes Reference

The following attributes are exported:

- **id** - The db subnet group name.
- **arn** - The ARN of the db subnet group.

## » Import

DB Subnet groups can be imported using the **name**, e.g.

```
$ terraform import aws_db_subnet_group.default production-subnet-group
```

## » aws\_\_rds\_\_cluster

Provides an RDS Cluster Resource. A Cluster Resource defines attributes that are applied to the entire cluster of RDS Cluster Instances. Use the RDS Cluster resource and RDS Cluster Instances to create and use Amazon Aurora, a MySQL-compatible database engine.

For more information on Amazon Aurora, see Aurora on Amazon RDS in the Amazon RDS User Guide.

Changes to a RDS Cluster can occur when you manually change a parameter, such as `port`, and are reflected in the next maintenance window. Because of this, Terraform may report a difference in its planning phase because a modification has not yet taken place. You can use the `apply_immediately` flag to instruct the service to apply the change immediately (see documentation below).

**Note:** using `apply_immediately` can result in a brief downtime as the server reboots. See the AWS Docs on RDS Maintenance for more information.

**Note:** All arguments including the username and password will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Example Usage

### » Aurora MySQL 2.x (MySQL 5.7)

```
resource "aws_rds_cluster" "default" {
  cluster_identifier      = "aurora-cluster-demo"
  engine                  = "aurora-mysql"
  availability_zones      = ["us-west-2a", "us-west-2b", "us-west-2c"]
  database_name           = "mydb"
  master_username         = "foo"
  master_password         = "bar"
  backup_retention_period = 5
  preferred_backup_window = "07:00-09:00"
}
```

### » Aurora MySQL 1.x (MySQL 5.6)

```
resource "aws_rds_cluster" "default" {
  cluster_identifier      = "aurora-cluster-demo"
  availability_zones      = ["us-west-2a", "us-west-2b", "us-west-2c"]
  database_name           = "mydb"
  master_username         = "foo"
  master_password         = "bar"
  backup_retention_period = 5
  preferred_backup_window = "07:00-09:00"
}
```

### » Aurora with PostgreSQL engine

```
resource "aws_rds_cluster" "postgresql" {
```

```

cluster_identifier      = "aurora-cluster-demo"
engine                  = "aurora-postgresql"
availability_zones      = ["us-west-2a", "us-west-2b", "us-west-2c"]
database_name          = "mydb"
master_username        = "foo"
master_password        = "bar"
backup_retention_period = 5
preferred_backup_window = "07:00-09:00"
}

```

## » Argument Reference

For more detailed documentation about each argument, refer to the AWS official documentation.

The following arguments are supported:

- **cluster\_identifier** - (Optional, Forces new resources) The cluster identifier. If omitted, Terraform will assign a random, unique identifier.
- **cluster\_identifier\_prefix** - (Optional, Forces new resource) Creates a unique cluster identifier beginning with the specified prefix. Conflicts with **cluster\_identifier**.
- **database\_name** - (Optional) Name for an automatically created database on cluster creation. There are different naming restrictions per database engine: [RDS Naming Constraints](#)
- **master\_password** - (Required unless a **snapshot\_identifier** is provided) Password for the master DB user. Note that this may show up in logs, and it will be stored in the state file. Please refer to the [RDS Naming Constraints](#)
- **master\_username** - (Required unless a **snapshot\_identifier** is provided) Username for the master DB user. Please refer to the [RDS Naming Constraints](#)
- **final\_snapshot\_identifier** - (Optional) The name of your final DB snapshot when this DB cluster is deleted. If omitted, no final snapshot will be made.
- **skip\_final\_snapshot** - (Optional) Determines whether a final DB snapshot is created before the DB cluster is deleted. If true is specified, no DB snapshot is created. If false is specified, a DB snapshot is created before the DB cluster is deleted, using the value from **final\_snapshot\_identifier**. Default is **false**.
- **availability\_zones** - (Optional) A list of EC2 Availability Zones that instances in the DB cluster can be created in
- **backup\_retention\_period** - (Optional) The days to retain backups for. Default 1
- **preferred\_backup\_window** - (Optional) The daily time range during which automated backups are created if automated backups are enabled

using the `BackupRetentionPeriod` parameter. Time in UTC Default: A 30-minute window selected at random from an 8-hour block of time per region. e.g. 04:00-09:00

- **preferred\_maintenance\_window** - (Optional) The weekly time range during which system maintenance can occur, in (UTC) e.g. wed:04:00-wed:04:30
- **port** - (Optional) The port on which the DB accepts connections
- **vpc\_security\_group\_ids** - (Optional) List of VPC security groups to associate with the Cluster
- **snapshot\_identifier** - (Optional) Specifies whether or not to create this cluster from a snapshot. You can use either the name or ARN when specifying a DB cluster snapshot, or the ARN when specifying a DB snapshot.
- **storage\_encrypted** - (Optional) Specifies whether the DB cluster is encrypted. The default is `false` if not specified.
- **replication\_source\_identifier** - (Optional) ARN of a source DB cluster or DB instance if this DB cluster is to be created as a Read Replica.
- **apply\_immediately** - (Optional) Specifies whether any cluster modifications are applied immediately, or during the next maintenance window. Default is `false`. See Amazon RDS Documentation for more information.
- **db\_subnet\_group\_name** - (Optional) A DB subnet group to associate with this DB instance. **NOTE:** This must match the `db_subnet_group_name` specified on every `aws_rds_cluster_instance` in the cluster.
- **db\_cluster\_parameter\_group\_name** - (Optional) A cluster parameter group to associate with the cluster.
- **kms\_key\_id** - (Optional) The ARN for the KMS encryption key. When specifying `kms_key_id`, `storage_encrypted` needs to be set to true.
- **iam\_roles** - (Optional) A List of ARNs for the IAM roles to associate to the RDS Cluster.
- **iam\_database\_authentication\_enabled** - (Optional) Specifies whether or mappings of AWS Identity and Access Management (IAM) accounts to database accounts is enabled.
- **engine** - (Optional) The name of the database engine to be used for this DB cluster. Defaults to `aurora`. Valid Values: `aurora`, `aurora-mysql`, `aurora-postgresql`
- **engine\_version** - (Optional) The database engine version.
- **source\_region** - (Optional) The source region for an encrypted replica DB cluster.

### » S3 Import Options

Full details on the core parameters and impacts are in the API Docs: `RestoreDBClusterFromS3`. Requires that the S3 bucket be in the same region as the RDS cluster you're trying to create. Sample

```

resource "aws_rds_cluster" "db" {
  engine = "aurora"

  s3_import {
    source_engine = "mysql"
    source_engine_version = "5.6"
    bucket_name = "mybucket"
    bucket_prefix = "backups"
    ingestion_role = "arn:aws:iam::1234567890:role/role-xtrabackup-rds-restore"
  }
}

```

- **bucket\_name** - (Required) The bucket name where your backup is stored
- **bucket\_prefix** - (Optional) Can be blank, but is the path to your backup
- **ingestion\_role** - (Required) Role applied to load the data.
- **source\_engine** - (Required) Source engine for the backup
- **source\_engine\_version** - (Required) Version of the source engine used to make the backup

This will not recreate the resource if the S3 object changes in some way. It's only used to initialize the database. This only works currently with the aurora engine. See AWS for currently supported engines and options. [Aurora S3 Migration Docs](#)

## » Attributes Reference

The following attributes are exported:

- **id** - The RDS Cluster Identifier
- **cluster\_identifier** - The RDS Cluster Identifier
- **cluster\_resource\_id** - The RDS Cluster Resource ID
- **cluster\_members** - List of RDS Instances that are a part of this cluster
- **allocated\_storage** - The amount of allocated storage
- **availability\_zones** - The availability zone of the instance
- **backup\_retention\_period** - The backup retention period
- **preferred\_backup\_window** - The daily time range during which the backups happen
- **preferred\_maintenance\_window** - The maintenance window
- **endpoint** - The DNS address of the RDS instance
- **reader\_endpoint** - A read-only endpoint for the Aurora cluster, automatically load-balanced across replicas
- **engine** - The database engine
- **engine\_version** - The database engine version
- **maintenance\_window** - The instance maintenance window
- **database\_name** - The database name
- **port** - The database port

- `status` - The RDS instance status
- `master_username` - The master username for the database
- `storage_encrypted` - Specifies whether the DB cluster is encrypted
- `replication_source_identifier` - ARN of the source DB cluster or DB instance if this DB cluster is created as a Read Replica.
- `hosted_zone_id` - The Route53 Hosted Zone ID of the endpoint

## » Timeouts

`aws_rds_cluster` provides the following Timeouts configuration options:

- `create` - (Default 120 minutes) Used for Cluster creation
- `update` - (Default 120 minutes) Used for Cluster modifications
- `delete` - (Default 120 minutes) Used for destroying cluster. This includes any cleanup task during the destroying process.

## » Import

RDS Clusters can be imported using the `cluster_identifier`, e.g.

```
$ terraform import aws_rds_cluster.aurora_cluster aurora-prod-cluster
```

## » `aws_rds_cluster_instance`

Provides an RDS Cluster Resource Instance. A Cluster Instance Resource defines attributes that are specific to a single instance in a RDS Cluster, specifically running Amazon Aurora.

Unlike other RDS resources that support replication, with Amazon Aurora you do not designate a primary and subsequent replicas. Instead, you simply add RDS Instances and Aurora manages the replication. You can use the count meta-parameter to make multiple instances and join them all to the same RDS Cluster, or you may specify different Cluster Instance resources with various `instance_class` sizes.

For more information on Amazon Aurora, see Aurora on Amazon RDS in the Amazon RDS User Guide.

## » Example Usage

```
resource "aws_rds_cluster_instance" "cluster_instances" {
  count      = 2
  identifier = "aurora-cluster-demo-${count.index}"
}
```



```

    cluster_identifier = "${aws_rds_cluster.default.id}"
    instance_class      = "db.r3.large"
}

resource "aws_rds_cluster" "default" {
  cluster_identifier = "aurora-cluster-demo"
  availability_zones = ["us-west-2a", "us-west-2b", "us-west-2c"]
  database_name      = "mydb"
  master_username    = "foo"
  master_password    = "barbut8chars"
}

```

## » Argument Reference

For more detailed documentation about each argument, refer to the AWS official documentation.

The following arguments are supported:

- **identifier** - (Optional, Forces new resource) The identifier for the RDS instance, if omitted, Terraform will assign a random, unique identifier.
- **identifier\_prefix** - (Optional, Forces new resource) Creates a unique identifier beginning with the specified prefix. Conflicts with **identifier**.
- **cluster\_identifier** - (Required) The identifier of the `aws_rds_cluster` in which to launch this instance.
- **engine** - (Optional) The name of the database engine to be used for the RDS instance. Defaults to `aurora`. Valid Values: `aurora`, `aurora-mysql`, `aurora-postgresql`.
- **engine\_version** - (Optional) The database engine version.
- **instance\_class** - (Required) The instance class to use. For details on CPU and memory, see [Scaling Aurora DB Instances](#). Aurora currently supports the below instance classes.
  - `db.t2.small`
  - `db.t2.medium`
  - `db.r3.large`
  - `db.r3.xlarge`
  - `db.r3.2xlarge`
  - `db.r3.4xlarge`
  - `db.r3.8xlarge`
  - `db.r4.large`
  - `db.r4.xlarge`
  - `db.r4.2xlarge`
  - `db.r4.4xlarge`
  - `db.r4.8xlarge`
  - `db.r4.16xlarge`

- **publicly\_accessible** - (Optional) Bool to control if instance is publicly accessible. Default **false**. See the documentation on Creating DB Instances for more details on controlling this property.
- **db\_subnet\_group\_name** - (Required if **publicly\_accessible** = **false**, Optional otherwise) A DB subnet group to associate with this DB instance. **NOTE:** This must match the **db\_subnet\_group\_name** of the attached **aws\_rds\_cluster**.
- **db\_parameter\_group\_name** - (Optional) The name of the DB parameter group to associate with this instance.
- **apply\_immediately** - (Optional) Specifies whether any database modifications are applied immediately, or during the next maintenance window. Default is **false**.
- **monitoring\_role\_arn** - (Optional) The ARN for the IAM role that permits RDS to send enhanced monitoring metrics to CloudWatch Logs. You can find more information on the AWS Documentation what IAM permissions are needed to allow Enhanced Monitoring for RDS Instances.
- **monitoring\_interval** - (Optional) The interval, in seconds, between points when Enhanced Monitoring metrics are collected for the DB instance. To disable collecting Enhanced Monitoring metrics, specify 0. The default is 0. Valid Values: 0, 1, 5, 10, 15, 30, 60.
- **promotion\_tier** - (Optional) Default 0. Failover Priority setting on instance level. The reader who has lower tier has higher priority to get promoter to writer.
- **availability\_zone** - (Optional, Computed) The EC2 Availability Zone that the DB instance is created in. See docs about the details.
- **preferred\_backup\_window** - (Optional) The daily time range during which automated backups are created if automated backups are enabled. Eg: "04:00-09:00"
- **preferred\_maintenance\_window** - (Optional) The window to perform maintenance in. Syntax: "ddd:hh24:mi-ddd:hh24:mi". Eg: "Mon:00:00-Mon:03:00".
- **auto\_minor\_version\_upgrade** - (Optional) Indicates that minor engine upgrades will be applied automatically to the DB instance during the maintenance window. Default **true**.
- **performance\_insights\_enabled** - (Optional) Specifies whether Performance Insights is enabled or not.
- **performance\_insights\_kms\_key\_id** - (Optional) The ARN for the KMS key to encrypt Performance Insights data. When specifying **performance\_insights\_kms\_key\_id**, **performance\_insights\_enabled** needs to be set to **true**.
- **tags** - (Optional) A mapping of tags to assign to the instance.

## » Attributes Reference

The following attributes are exported:

- `cluster_identifier` - The RDS Cluster Identifier
- `identifier` - The Instance identifier
- `id` - The Instance identifier
- `writer` - Boolean indicating if this instance is writable. **False** indicates this instance is a read replica.
- `allocated_storage` - The amount of allocated storage
- `availability_zone` - The availability zone of the instance
- `endpoint` - The DNS address for this instance. May not be writable
- `engine` - The database engine
- `engine_version` - The database engine version
- `database_name` - The database name
- `port` - The database port
- `status` - The RDS instance status
- `storage_encrypted` - Specifies whether the DB cluster is encrypted.
- `kms_key_id` - The ARN for the KMS encryption key if one is set to the cluster.
- `dbi_resource_id` - The region-unique, immutable identifier for the DB instance.
- `performance_insights_enabled` - Specifies whether Performance Insights is enabled or not.
- `performance_insights_kms_key_id` - The ARN for the KMS encryption key used by Performance Insights.

## » Timeouts

`aws_rds_cluster_instance` provides the following Timeouts configuration options:

- `create` - (Default 90 minutes) Used for Creating Instances, Replicas, and restoring from Snapshots
- `update` - (Default 90 minutes) Used for Database modifications
- `delete` - (Default 90 minutes) Used for destroying databases. This includes the time required to take snapshots

## » Import

RDS Cluster Instances can be imported using the `identifier`, e.g.

```
$ terraform import aws_rds_cluster_instance.prod_instance_1 aurora-cluster-instance-1
```

## » `aws_rds_cluster_parameter_group`

Provides an RDS DB cluster parameter group resource.

## » Example Usage

```
resource "aws_rds_cluster_parameter_group" "default" {
  name      = "rds-cluster-pg"
  family    = "aurora5.6"
  description = "RDS default cluster parameter group"

  parameter {
    name = "character_set_server"
    value = "utf8"
  }

  parameter {
    name = "character_set_client"
    value = "utf8"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional, Forces new resource) The name of the DB cluster parameter group. If omitted, Terraform will assign a random, unique name.
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **family** - (Required) The family of the DB cluster parameter group.
- **description** - (Optional) The description of the DB cluster parameter group. Defaults to "Managed by Terraform".
- **parameter** - (Optional) A list of DB parameters to apply. Note that parameters may differ from a family to another. Full list of all parameters can be discovered via `aws rds describe-db-cluster-parameters` after initial creation of the group.
- **tags** - (Optional) A mapping of tags to assign to the resource.

Parameter blocks support the following:

- **name** - (Required) The name of the DB parameter.
- **value** - (Required) The value of the DB parameter.
- **apply\_method** - (Optional) "immediate" (default), or "pending-reboot". Some engines can't apply some parameters without a reboot, and you will need to specify "pending-reboot" here.

## » Attributes Reference

The following attributes are exported:

- `id` - The db cluster parameter group name.
- `arn` - The ARN of the db cluster parameter group.

## » Import

RDS Cluster Parameter Groups can be imported using the `name`, e.g.

```
$ terraform import aws_rds_cluster_parameter_group.cluster_pg production-pg-1
```

## » `aws__redshift__cluster`

Provides a Redshift Cluster Resource.

**Note:** All arguments including the username and password will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Example Usage

```
resource "aws_redshift_cluster" "default" {
  cluster_identifier = "tf-redshift-cluster"
  database_name      = "mydb"
  master_username    = "foo"
  master_password    = "Mustbe8characters"
  node_type          = "dc1.large"
  cluster_type       = "single-node"
}
```

## » Argument Reference

For more detailed documentation about each argument, refer to the AWS official documentation.

The following arguments are supported:

- `cluster_identifier` - (Required) The Cluster Identifier. Must be a lower case string.
- `database_name` - (Optional) The name of the first database to be created when the cluster is created. If you do not provide a name, Amazon Redshift will create a default database called `dev`.

- **node\_type** - (Required) The node type to be provisioned for the cluster.
- **cluster\_type** - (Optional) The cluster type to use. Either **single-node** or **multi-node**.
- **master\_password** - (Required unless a **snapshot\_identifier** is provided) Password for the master DB user. Note that this may show up in logs, and it will be stored in the state file. Password must contain at least 8 chars and contain at least one uppercase letter, one lowercase letter, and one number.
- **master\_username** - (Required unless a **snapshot\_identifier** is provided) Username for the master DB user.
- **cluster\_security\_groups** - (Optional) A list of security groups to be associated with this cluster.
- **vpc\_security\_group\_ids** - (Optional) A list of Virtual Private Cloud (VPC) security groups to be associated with the cluster.
- **cluster\_subnet\_group\_name** - (Optional) The name of a cluster subnet group to be associated with this cluster. If this parameter is not provided the resulting cluster will be deployed outside virtual private cloud (VPC).
- **availability\_zone** - (Optional) The EC2 Availability Zone (AZ) in which you want Amazon Redshift to provision the cluster. For example, if you have several EC2 instances running in a specific Availability Zone, then you might want the cluster to be provisioned in the same zone in order to decrease network latency.
- **preferred\_maintenance\_window** - (Optional) The weekly time range (in UTC) during which automated cluster maintenance can occur. Format: ddd:hh24:mi-ddd:hh24:mi
- **cluster\_parameter\_group\_name** - (Optional) The name of the parameter group to be associated with this cluster.
- **automated\_snapshot\_retention\_period** - (Optional) The number of days that automated snapshots are retained. If the value is 0, automated snapshots are disabled. Even if automated snapshots are disabled, you can still create manual snapshots when you want with `create-cluster-snapshot`. Default is 1.
- **port** - (Optional) The port number on which the cluster accepts incoming connections. The cluster is accessible only via the JDBC and ODBC connection strings. Part of the connection string requires the port on which the cluster will listen for incoming connections. Default port is 5439.
- **cluster\_version** - (Optional) The version of the Amazon Redshift engine software that you want to deploy on the cluster. The version selected runs on all the nodes in the cluster.

- **allow\_version\_upgrade** - (Optional) If true , major version upgrades can be applied during the maintenance window to the Amazon Redshift engine that is running on the cluster. Default is true
- **number\_of\_nodes** - (Optional) The number of compute nodes in the cluster. This parameter is required when the **ClusterType** parameter is specified as multi-node. Default is 1.
- **publicly\_accessible** - (Optional) If true, the cluster can be accessed from a public network. Default is **true**.
- **encrypted** - (Optional) If true , the data in the cluster is encrypted at rest.
- **enhanced\_vpc\_routing** - (Optional) If true , enhanced VPC routing is enabled.
- **kms\_key\_id** - (Optional) The ARN for the KMS encryption key. When specifying **kms\_key\_id**, **encrypted** needs to be set to true.
- **elastic\_ip** - (Optional) The Elastic IP (EIP) address for the cluster.
- **skip\_final\_snapshot** - (Optional) Determines whether a final snapshot of the cluster is created before Amazon Redshift deletes the cluster. If true , a final cluster snapshot is not created. If false , a final cluster snapshot is created before the cluster is deleted. Default is false.
- **final\_snapshot\_identifier** - (Optional) The identifier of the final snapshot that is to be created immediately before deleting the cluster. If this parameter is provided, **skip\_final\_snapshot** must be false.
- **snapshot\_identifier** - (Optional) The name of the snapshot from which to create the new cluster.
- **snapshot\_cluster\_identifier** - (Optional) The name of the cluster the source snapshot was created from.
- **owner\_account** - (Optional) The AWS customer account used to create or copy the snapshot. Required if you are restoring a snapshot you do not own, optional if you own the snapshot.
- **iam\_roles** - (Optional) A list of IAM Role ARNs to associate with the cluster. A Maximum of 10 can be associated to the cluster at any time.
- **logging** - (Optional) Logging, documented below.
- **snapshot\_copy** - (Optional) Configuration of automatic copy of snapshots from one region to another. Documented below.
- **tags** - (Optional) A mapping of tags to assign to the resource.

## » Nested Blocks

### » logging

- **enable** - (Required) Enables logging information such as queries and connection attempts, for the specified Amazon Redshift cluster.
- **bucket\_name** - (Optional, required when **enable\_logging** is **true**) The name of an existing S3 bucket where the log files are to be stored. Must be in the same region as the cluster and the cluster must have read bucket and put object permissions. For more information on the permissions required for the bucket, please read the AWS documentation
- **s3\_key\_prefix** - (Optional) The prefix applied to the log file names.

### » snapshot\_copy

- **destination\_region** - (Required) The destination region that you want to copy snapshots to.
- **retention\_period** - (Optional) The number of days to retain automated snapshots in the destination region after they are copied from the source region. Defaults to 7.
- **grant\_name** - (Optional) The name of the snapshot copy grant to use when snapshots of an AWS KMS-encrypted cluster are copied to the destination region.

## » Attributes Reference

The following attributes are exported:

- **id** - The Redshift Cluster ID.
- **cluster\_identifier** - The Cluster Identifier
- **cluster\_type** - The cluster type
- **node\_type** - The type of nodes in the cluster
- **database\_name** - The name of the default database in the Cluster
- **availability\_zone** - The availability zone of the Cluster
- **automated\_snapshot\_retention\_period** - The backup retention period
- **preferred\_maintenance\_window** - The backup window
- **endpoint** - The connection endpoint
- **encrypted** - Whether the data in the cluster is encrypted
- **cluster\_security\_groups** - The security groups associated with the cluster
- **vpc\_security\_group\_ids** - The VPC security group Ids associated with the cluster
- **port** - The Port the cluster responds on
- **cluster\_version** - The version of Redshift engine software
- **cluster\_parameter\_group\_name** - The name of the parameter group to be associated with this cluster



- `cluster_subnet_group_name` - The name of a cluster subnet group to be associated with this cluster
- `cluster_public_key` - The public key for the cluster
- `cluster_revision_number` - The specific revision number of the database in the cluster

## » Import

Redshift Clusters can be imported using the `cluster_identifier`, e.g.

```
$ terraform import aws_redshift_cluster.myprodcluster tf-redshift-cluster-12345
```

## » `aws_redshift_parameter_group`

Provides a Redshift Cluster parameter group resource.

## » Example Usage

```
resource "aws_redshift_parameter_group" "bar" {
  name      = "parameter-group-test-terraform"
  family    = "redshift-1.0"

  parameter {
    name = "require_ssl"
    value = "true"
  }

  parameter {
    name = "query_group"
    value = "example"
  }

  parameter {
    name = "enable_user_activity_logging"
    value = "true"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the Redshift parameter group.
- **family** - (Required) The family of the Redshift parameter group.
- **description** - (Optional) The description of the Redshift parameter group. Defaults to "Managed by Terraform".
- **parameter** - (Optional) A list of Redshift parameters to apply.

Parameter blocks support the following:

- **name** - (Required) The name of the Redshift parameter.
- **value** - (Required) The value of the Redshift parameter.

You can read more about the parameters that Redshift supports in the documentation

## » Attributes Reference

The following attributes are exported:

- **id** - The Redshift parameter group name.

## » Import

Redshift Parameter Groups can be imported using the **name**, e.g.

```
$ terraform import aws_redshift_parameter_group.paramgroup1 parameter-group-test-terraform
```

## » `aws__redshift__security__group`

Creates a new Amazon Redshift security group. You use security groups to control access to non-VPC clusters

## » Example Usage

```
resource "aws_redshift_security_group" "default" {
  name = "redshift-sg"

  ingress {
    cidr = "10.0.0.0/24"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the Redshift security group.
- **description** - (Optional) The description of the Redshift security group. Defaults to "Managed by Terraform".
- **ingress** - (Optional) A list of ingress rules.

Ingress blocks support the following:

- **cidr** - The CIDR block to accept
- **security\_group\_name** - The name of the security group to authorize
- **security\_group\_owner\_id** - The owner Id of the security group provided by **security\_group\_name**.

## » Attributes Reference

The following attributes are exported:

- **id** - The Redshift security group ID.

## » Import

Redshift security groups can be imported using the **name**, e.g.

```
$ terraform import aws_redshift_security_group.testgroup1 redshift_test_group
```

## » `aws__redshift__subnet__group`

Creates a new Amazon Redshift subnet group. You must provide a list of one or more subnets in your existing Amazon Virtual Private Cloud (Amazon VPC) when creating Amazon Redshift subnet group.

## » Example Usage

```
resource "aws_vpc" "foo" {
  cidr_block = "10.1.0.0/16"
}

resource "aws_subnet" "foo" {
  cidr_block          = "10.1.1.0/24"
  availability_zone   = "us-west-2a"
  vpc_id              = "${aws_vpc.foo.id}"
}
```

```

    tags {
      Name = "tf-dbsubnet-test-1"
    }
  }

resource "aws_subnet" "bar" {
  cidr_block      = "10.1.2.0/24"
  availability_zone = "us-west-2b"
  vpc_id          = "${aws_vpc.foo.id}"

  tags {
    Name = "tf-dbsubnet-test-2"
  }
}

resource "aws_redshift_subnet_group" "foo" {
  name          = "foo"
  subnet_ids    = ["${aws_subnet.foo.id}", "${aws_subnet.bar.id}"]

  tags {
    environment = "Production"
  }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the Redshift Subnet group.
- **description** - (Optional) The description of the Redshift Subnet group. Defaults to "Managed by Terraform".
- **subnet\_ids** - (Required) An array of VPC subnet IDs.
- **tags** - (Optional) A mapping of tags to assign to the resource.

## » Attributes Reference

The following attributes are exported:

- **id** - The Redshift Subnet group ID.

## » Import

Redshift subnet groups can be imported using the **name**, e.g.

```
$ terraform import aws_redshift_subnet_group.testgroup1 test-cluster-subnet-group
```

## » **aws\_waf\_byte\_match\_set**

Provides a WAF Byte Match Set Resource

### » **Example Usage**

```
resource "aws_waf_byte_match_set" "byte_set" {
  name = "tf_waf_byte_match_set"

  byte_match_tuples {
    text_transformation    = "NONE"
    target_string           = "badreferer"
    positional_constraint = "CONTAINS"

    field_to_match {
      type = "HEADER"
      data = "referer"
    }
  }
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) The name or description of the Byte Match Set.
- **byte\_match\_tuples** - Specifies the bytes (typically a string that corresponds with ASCII characters) that you want to search for in web requests, the location in requests that you want to search, and other settings.

### » **Nested blocks**

#### » **byte\_match\_tuples**

#### » **Arguments**

- **field\_to\_match** - (Required) The part of a web request that you want to search, such as a specified header or a query string.

- **positional\_constraint** - (Required) Within the portion of a web request that you want to search (for example, in the query string, if any), specify where you want to search. e.g. **CONTAINS**, **CONTAINS\_WORD** or **EXACTLY**. See docs for all supported values.
- **target\_string** - (Optional) The value that you want to search for. e.g. **HEADER**, **METHOD** or **BODY**. See docs for all supported values.
- **text\_transformation** - (Required) Text transformations used to eliminate unusual formatting that attackers use in web requests in an effort to bypass AWS WAF. If you specify a transformation, AWS WAF performs the transformation on **target\_string** before inspecting a request for a match. e.g. **CMD\_LINE**, **HTML\_ENTITY\_DECODE** or **NONE**. See docs for all supported values.

» **field\_to\_match**

» **Arguments**

- **data** - (Optional) When **type** is **HEADER**, enter the name of the header that you want to search, e.g. **User-Agent** or **Referer**. If **type** is any other value, omit this field.
- **type** - (Required) The part of the web request that you want AWS WAF to search for a specified string. e.g. **HEADER**, **METHOD** or **BODY**. See docs for all supported values.

» **Remarks**

» **Attributes Reference**

The following attributes are exported:

- **id** - The ID of the WAF Byte Match Set.

» **aws\_waf\_geo\_match\_set**

Provides a WAF Geo Match Set Resource

» **Example Usage**

```
resource "aws_waf_geo_match_set" "geo_match_set" {
  name = "geo_match_set"

  geo_match_constraint {
```

```

    type = "Country"
    value = "US"
  }

  geo_match_constraint {
    type = "Country"
    value = "CA"
  }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name or description of the GeoMatchSet.
- **geo\_match\_constraint** - (Optional) The GeoMatchConstraint objects which contain the country that you want AWS WAF to search for.

## » Nested Blocks

### » `geo_match_constraint`

#### » Arguments

- **type** - (Required) The type of geographical area you want AWS WAF to search for. Currently Country is the only valid value.
- **value** - (Required) The country that you want AWS WAF to search for. This is the two-letter country code, e.g. **US**, **CA**, **RU**, **CN**, etc. See docs for all supported values.

## » Remarks

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF GeoMatchSet.

## » `aws_waf_ipset`

Provides a WAF IPSet Resource

## » Example Usage

```
resource "aws_waf_ipset" "ipset" {  
  name = "tfIPSet"  
  
  ip_set_descriptors {  
    type = "IPv4"  
    value = "192.0.7.0/24"  
  }  
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name or description of the IPSet.
- **ip\_set\_descriptors** - (Optional) Specifies the IP address type (IPv4 or IPv6) and the IP address range (in CIDR format) that web requests originate from.

## » Nested Blocks

### » ip\_set\_descriptors

#### » Arguments

- **type** - (Required) Type of the IP address - **IPv4** or **IPv6**.
- **value** - (Required) An IPv4 or IPv6 address specified via CIDR notation.  
e.g. 192.0.2.44/32 or 1111:0000:0000:0000:0000:0000:0000/64

## » Remarks

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF IPSet.

## » aws\_waf\_rate\_based\_rule

Provides a WAF Rate Based Rule Resource



## » Example Usage

```
resource "aws_waf_ipset" "ipset" {
  name = "tfIPSet"

  ip_set_descriptors {
    type = "IPv4"
    value = "192.0.7.0/24"
  }
}

resource "aws_waf_rate_based_rule" "wafrule" {
  depends_on = ["aws_waf_ipset.ipset"]
  name       = "tfWAFRule"
  metric_name = "tfWAFRule"

  rate_key = "IP"
  rate_limit = 2000

  predicates {
    data_id = "${aws_waf_ipset.ipset.id}"
    negated = false
    type    = "IPMatch"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **metric\_name** - (Required) The name or description for the Amazon Cloud-Watch metric of this rule.
- **name** - (Required) The name or description of the rule.
- **rate\_key** - (Required) Valid value is IP.
- **rate\_limit** - (Required) The maximum number of requests, which have an identical value in the field specified by the RateKey, allowed in a five-minute period. Minimum value is 2000.
- **predicates** - (Optional) One of ByteMatchSet, IPSet, SizeConstraintSet, SqlInjectionMatchSet, or XssMatchSet objects to include in a rule.

## » Nested Blocks

### » predicates

See the WAF Documentation for more information.

### » Arguments

- **negated** - (Required) Set this to **false** if you want to allow, block, or count requests based on the settings in the specified `ByteMatchSet`, `IPSet`, `SqlInjectionMatchSet`, `XssMatchSet`, or `SizeConstraintSet`. For example, if an `IPSet` includes the IP address 192.0.2.44, AWS WAF will allow or block requests based on that IP address. If set to **true**, AWS WAF will allow, block, or count requests based on all IP addresses *except* 192.0.2.44.
- **data\_id** - (Required) A unique identifier for a predicate in the rule, such as Byte Match Set ID or IPSet ID.
- **type** - (Required) The type of predicate in a rule. Valid values: `ByteMatch`, `GeoMatch`, `IPMatch`, `RegexMatch`, `SizeConstraint`, `SqlInjectionMatch`, or `XssMatch`.

## » Remarks

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF rule.

## » `aws_waf_regex_match_set`

Provides a WAF Regex Match Set Resource

## » Example Usage

```
resource "aws_waf_regex_match_set" "example" {
  name = "example"
  regex_match_tuple {
    field_to_match {
      data = "User-Agent"
      type = "HEADER"
    }
  }
}
```

```

        regex_pattern_set_id = "${aws_waf_regex_pattern_set.example.id}"
        text_transformation = "NONE"
    }
}

resource "aws_waf_regex_pattern_set" "example" {
    name = "example"
    regex_pattern_strings = ["one", "two"]
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name or description of the Regex Match Set.
- **regex\_match\_tuple** - (Required) The regular expression pattern that you want AWS WAF to search for in web requests, the location in requests that you want AWS WAF to search, and other settings. See below.

## » Nested Arguments

### » **regex\_match\_tuple**

- **field\_to\_match** - (Required) The part of a web request that you want to search, such as a specified header or a query string.
- **regex\_pattern\_set\_id** - (Required) The ID of a Regex Pattern Set.
- **text\_transformation** - (Required) Text transformations used to eliminate unusual formatting that attackers use in web requests in an effort to bypass AWS WAF. e.g. `CMD_LINE`, `HTML_ENTITY_DECODE` or `NONE`. See docs for all supported values.

### » **field\_to\_match**

- **data** - (Optional) When **type** is `HEADER`, enter the name of the header that you want to search, e.g. `User-Agent` or `Referer`. If **type** is any other value, omit this field.
- **type** - (Required) The part of the web request that you want AWS WAF to search for a specified string. e.g. `HEADER`, `METHOD` or `BODY`. See docs for all supported values.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF Regex Match Set.

## » **aws\_waf\_regex\_pattern\_set**

Provides a WAF Regex Pattern Set Resource

### » **Example Usage**

```
resource "aws_waf_regex_pattern_set" "example" {  
  name = "tf_waf_regex_pattern_set"  
  regex_pattern_strings = ["one", "two"]  
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) The name or description of the Regex Pattern Set.
- **regex\_pattern\_strings** - (Optional) A list of regular expression (regex) patterns that you want AWS WAF to search for, such as B[a@]dB[o]t.

### » **Attributes Reference**

The following attributes are exported:

- **id** - The ID of the WAF Regex Pattern Set.

## » **aws\_waf\_rule**

Provides a WAF Rule Resource

### » **Example Usage**

```
resource "aws_waf_ipset" "ipset" {  
  name = "tfIPSet"  
  
  ip_set_descriptors {  
    type = "IPv4"  
    value = "192.0.7.0/24"  
  }  
}  
  
resource "aws_waf_rule" "wafrule" {
```

```

depends_on = ["aws_waf_ipset.ipset"]
name      = "tfWAFRule"
metric_name = "tfWAFRule"

predicates {
  data_id = "${aws_waf_ipset.ipset.id}"
  negated = false
  type    = "IPMatch"
}
}

```

## » Argument Reference

The following arguments are supported:

- **metric\_name** - (Required) The name or description for the Amazon Cloud-Watch metric of this rule.
- **name** - (Required) The name or description of the rule.
- **predicates** - (Optional) One of ByteMatchSet, IPSet, SizeConstraintSet, SqlInjectionMatchSet, or XssMatchSet objects to include in a rule.

## » Nested Blocks

### » predicates

See the WAF Documentation for more information.

### » Arguments

- **negated** - (Required) Set this to **false** if you want to allow, block, or count requests based on the settings in the specified `waf_byte_match_set`, `waf_ipset`, `aws_waf_size_constraint_set`, `aws_waf_sql_injection_match_set` or `aws_waf_xss_match_set`. For example, if an IPSet includes the IP address `192.0.2.44`, AWS WAF will allow or block requests based on that IP address. If set to **true**, AWS WAF will allow, block, or count requests based on all IP addresses *except* `192.0.2.44`.
- **data\_id** - (Required) A unique identifier for a predicate in the rule, such as Byte Match Set ID or IPSet ID.
- **type** - (Required) The type of predicate in a rule. Valid values: `ByteMatch`, `GeoMatch`, `IPMatch`, `RegexMatch`, `SizeConstraint`, `SqlInjectionMatch`, or `XssMatch`.

## » Remarks

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the WAF rule.

## » `aws_waf_rule_group`

Provides a WAF Rule Group Resource

## » Example Usage

```
resource "aws_waf_rule" "example" {
  name          = "example"
  metric_name   = "example"
}

resource "aws_waf_rule_group" "example" {
  name = "example"
  metric_name = "example"
  activated_rule {
    action {
      type = "COUNT"
    }
  }
  priority = 50
  rule_id = "${aws_waf_rule.example.id}"
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) A friendly name of the rule group
- `metric_name` - (Required) A friendly name for the metrics from the rule group
- `activated_rule` - (Optional) A list of activated rules, see below

## » Nested Blocks

### » `activated_rule`

### » Arguments

- `action` - (Required) Specifies the action that CloudFront or AWS WAF takes when a web request matches the conditions in the rule.
  - `type` - (Required) e.g. `BLOCK`, `ALLOW`, or `COUNT`
- `priority` - (Required) Specifies the order in which the rules are evaluated. Rules with a lower value are evaluated before rules with a higher value.
- `rule_id` - (Required) The ID of a rule
- `type` - (Optional) The rule type, either `REGULAR`, `[RATE_BASED]`([/docs/providers/aws/r/waf\\_rate\\_based\\_rule\\_group.html](/docs/providers/aws/r/waf_rate_based_rule_group.html)) or `GROUP`. Defaults to `REGULAR`.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the WAF rule group.

## » `aws_waf_size_constraint_set`

Provides a WAF Size Constraint Set Resource

## » Example Usage

```
resource "aws_waf_size_constraint_set" "size_constraint_set" {
  name = "tfsize_constraints"

  size_constraints {
    text_transformation = "NONE"
    comparison_operator = "EQ"
    size                = "4096"

    field_to_match {
      type = "BODY"
    }
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name or description of the Size Constraint Set.
- **size\_constraints** - (Optional) Specifies the parts of web requests that you want to inspect the size of.

## » Nested Blocks

### » **size\_constraints**

#### » Arguments

- **field\_to\_match** - (Required) Specifies where in a web request to look for the size constraint.
- **comparison\_operator** - (Required) The type of comparison you want to perform. e.g. `EQ`, `NE`, `LT`, `GT`. See docs for all supported values.
- **size** - (Required) The size in bytes that you want to compare against the size of the specified **field\_to\_match**. Valid values are between 0 - 21474836480 bytes (0 - 20 GB).
- **text\_transformation** - (Required) Text transformations used to eliminate unusual formatting that attackers use in web requests in an effort to bypass AWS WAF. If you specify a transformation, AWS WAF performs the transformation on **field\_to\_match** before inspecting a request for a match. e.g. `CMD_LINE`, `HTML_ENTITY_DECODE` or `NONE`. See docs for all supported values. **Note:** if you choose `BODY` as **type**, you must choose `NONE` because CloudFront forwards only the first 8192 bytes for inspection.

### » **field\_to\_match**

#### » Arguments

- **data** - (Optional) When **type** is `HEADER`, enter the name of the header that you want to search, e.g. `User-Agent` or `Referer`. If **type** is any other value, omit this field.
- **type** - (Required) The part of the web request that you want AWS WAF to search for a specified string. e.g. `HEADER`, `METHOD` or `BODY`. See docs for all supported values.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF Size Constraint Set.



## » `aws_waf_sql_injection_match_set`

Provides a WAF SQL Injection Match Set Resource

### » Example Usage

```
resource "aws_waf_sql_injection_match_set" "sql_injection_match_set" {
  name = "tf-sql_injection_match_set"

  sql_injection_match_tuples {
    text_transformation = "URL_DECODE"

    field_to_match {
      type = "QUERY_STRING"
    }
  }
}
```

### » Argument Reference

The following arguments are supported:

- `name` - (Required) The name or description of the `SizeConstraintSet`.
- `sql_injection_match_tuples` - (Optional) The parts of web requests that you want AWS WAF to inspect for malicious SQL code and, if you want AWS WAF to inspect a header, the name of the header.

### » Nested Blocks

#### » `sql_injection_match_tuples`

- `field_to_match` - (Required) Specifies where in a web request to look for snippets of malicious SQL code.
- `text_transformation` - (Required) Text transformations used to eliminate unusual formatting that attackers use in web requests in an effort to bypass AWS WAF. If you specify a transformation, AWS WAF performs the transformation on `field_to_match` before inspecting a request for a match. e.g. `CMD_LINE`, `HTML_ENTITY_DECODE` or `NONE`. See docs for all supported values.

#### » `field_to_match`

#### » Arguments

- **data** - (Optional) When **type** is **HEADER**, enter the name of the header that you want to search, e.g. **User-Agent** or **Referer**. If **type** is any other value, omit this field.
- **type** - (Required) The part of the web request that you want AWS WAF to search for a specified string. e.g. **HEADER**, **METHOD** or **BODY**. See docs for all supported values.

## » Remarks

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF SQL Injection Match Set.

## » `aws_waf_web_acl`

Provides a WAF Web ACL Resource

## » Example Usage

```
resource "aws_waf_ipset" "ipset" {
  name = "tfIPSet"

  ip_set_descriptors {
    type = "IPV4"
    value = "192.0.7.0/24"
  }
}

resource "aws_waf_rule" "wafrule" {
  depends_on = ["aws_waf_ipset.ipset"]
  name       = "tfWAFRule"
  metric_name = "tfWAFRule"

  predicates {
    data_id = "${aws_waf_ipset.ipset.id}"
    negated = false
    type    = "IPMatch"
  }
}

resource "aws_waf_web_acl" "waf_acl" {
```

```

depends_on = ["aws_waf_ipset.ipset", "aws_waf_rule.wafrule"]
name      = "tfWebACL"
metric_name = "tfWebACL"

default_action {
  type = "ALLOW"
}

rules {
  action {
    type = "BLOCK"
  }

  priority = 1
  rule_id  = "${aws_waf_rule.wafrule.id}"
  type     = "REGULAR"
}
}

```

## » Argument Reference

The following arguments are supported:

- **default\_action** - (Required) The action that you want AWS WAF to take when a request doesn't match the criteria in any of the rules that are associated with the web ACL.
- **metric\_name** - (Required) The name or description for the Amazon CloudWatch metric of this web ACL.
- **name** - (Required) The name or description of the web ACL.
- **rules** - (Required) The rules to associate with the web ACL and the settings for each rule.

## » Nested Blocks

### » default\_action

### » Arguments

- **type** - (Required) Specifies how you want AWS WAF to respond to requests that match the settings in a rule. e.g. ALLOW, BLOCK or COUNT

### » rules

See docs for all details and supported values.

## » Arguments

- **action** - (Required) The action that CloudFront or AWS WAF takes when a web request matches the conditions in the rule. e.g. `ALLOW`, `BLOCK` or `COUNT`
- **priority** - (Required) Specifies the order in which the rules in a WebACL are evaluated. Rules with a lower value are evaluated before rules with a higher value.
- **rule\_id** - (Required) ID of the associated rule
- **type** - (Optional) The rule type, either `REGULAR`, as defined by Rule, or `RATE_BASED`, as defined by RateBasedRule. The default is `REGULAR`. If you add a `RATE_BASED` rule, you need to set **type** as `RATE_BASED`.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF WebACL.

## » `aws_waf_xss_match_set`

Provides a WAF XSS Match Set Resource

## » Example Usage

```
resource "aws_waf_xss_match_set" "xss_match_set" {
  name = "xss_match_set"

  xss_match_tuples {
    text_transformation = "NONE"

    field_to_match {
      type = "URI"
    }
  }

  xss_match_tuples {
    text_transformation = "NONE"

    field_to_match {
      type = "QUERY_STRING"
    }
  }
}
```

}

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name or description of the SizeConstraintSet.
- **xss\_match\_tuples** - (Optional) The parts of web requests that you want to inspect for cross-site scripting attacks.

## » Nested Blocks

### » **xss\_match\_tuples**

- **field\_to\_match** - (Required) Specifies where in a web request to look for cross-site scripting attacks.
- **text\_transformation** - (Required) Text transformations used to eliminate unusual formatting that attackers use in web requests in an effort to bypass AWS WAF. If you specify a transformation, AWS WAF performs the transformation on **target\_string** before inspecting a request for a match. e.g. **CMD\_LINE**, **HTML\_ENTITY\_DECODE** or **NONE**. See docs for all supported values.

### » **field\_to\_match**

### » Arguments

- **data** - (Optional) When **type** is **HEADER**, enter the name of the header that you want to search, e.g. **User-Agent** or **Referer**. If **type** is any other value, omit this field.
- **type** - (Required) The part of the web request that you want AWS WAF to search for a specified string. e.g. **HEADER**, **METHOD** or **BODY**. See docs for all supported values.

## » Remarks

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF XssMatchSet.

## » `aws_wafregional_byte_match_set`

Provides a WAF Regional Byte Match Set Resource for use with Application Load Balancer.

### » Example Usage

```
resource "aws_wafregional_byte_match_set" "byte_set" {
  name = "tf_waf_byte_match_set"
  byte_match_tuple {
    text_transformation = "NONE"
    target_string = "badrefer1"
    positional_constraint = "CONTAINS"
    field_to_match {
      type = "HEADER"
      data = "referer"
    }
  }
}
```

### » Argument Reference

The following arguments are supported:

- **name** - (Required) The name or description of the ByteMatchSet.
- **byte\_match\_tuple** - (Optional) Settings for the ByteMatchSet, such as the bytes (typically a string that corresponds with ASCII characters) that you want AWS WAF to search for in web requests. ByteMatchTuple documented below.

ByteMatchTuple(byte\_match\_tuple) support the following:

- **field\_to\_match** - (Required) Settings for the ByteMatchTuple. FieldToMatch documented below.
- **positional\_constraint** - (Required) Within the portion of a web request that you want to search.
- **target\_string** - (Required) The value that you want AWS WAF to search for. The maximum length of the value is 50 bytes.
- **text\_transformation** - (Required) The formatting way for web request.

FieldToMatch(field\_to\_match) support following:

- **data** - (Optional) When the value of Type is HEADER, enter the name of the header that you want AWS WAF to search, for example, User-Agent or Referer. If the value of Type is any other value, omit Data.

- **type** - (Required) The part of the web request that you want AWS WAF to search for a specified string.

## » Remarks

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF ByteMatchSet.

## » `aws_wafregional_geo_match_set`

Provides a WAF Regional Geo Match Set Resource

## » Example Usage

```
resource "aws_wafregional_geo_match_set" "geo_match_set" {
  name = "geo_match_set"

  geo_match_constraint {
    type = "Country"
    value = "US"
  }

  geo_match_constraint {
    type = "Country"
    value = "CA"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name or description of the Geo Match Set.
- **geo\_match\_constraint** - (Optional) The Geo Match Constraint objects which contain the country that you want AWS WAF to search for.

## » Nested Blocks

### » `geo_match_constraint`

#### » Arguments

- **type** - (Required) The type of geographical area you want AWS WAF to search for. Currently Country is the only valid value.
- **value** - (Required) The country that you want AWS WAF to search for. This is the two-letter country code, e.g. **US**, **CA**, **RU**, **CN**, etc. See docs for all supported values.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF Regional Geo Match Set.

## » `aws__wafregional_ipset`

Provides a WAF Regional IPSet Resource for use with Application Load Balancer.

## » Example Usage

```
resource "aws_wafregional_ipset" "ipset" {
  name = "tfIPSet"
  ip_set_descriptor {
    type = "IPV4"
    value = "192.0.7.0/24"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name or description of the IPSet.
- **ip\_set\_descriptor** - (Optional) The IP address type and IP address range (in CIDR notation) from which web requests originate.

IPSetDescriptor(ip\_set\_descriptor) support following:

- **type** - (Required) The string like IPV4 or IPV6.



- `value` - (Required) The CIDR notation.

## » Remarks

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the WAF IPSet.

## » `aws_wafregional_rate_based_rule`

Provides a WAF Rate Based Rule Resource

## » Example Usage

```
resource "aws_wafregional_ipset" "ipset" {
  name = "tfIPSet"

  ip_set_descriptors {
    type = "IPv4"
    value = "192.0.7.0/24"
  }
}

resource "aws_wafregional_rate_based_rule" "wafrule" {
  depends_on = ["aws_wafregional_ipset.ipset"]
  name       = "tfWAFRule"
  metric_name = "tfWAFRule"

  rate_key = "IP"
  rate_limit = 2000

  predicate {
    data_id = "${aws_wafregional_ipset.ipset.id}"
    negated = false
    type    = "IPMatch"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **metric\_name** - (Required) The name or description for the Amazon Cloud-Watch metric of this rule.
- **name** - (Required) The name or description of the rule.
- **rate\_key** - (Required) Valid value is IP.
- **rate\_limit** - (Required) The maximum number of requests, which have an identical value in the field specified by the RateKey, allowed in a five-minute period. Minimum value is 2000.
- **predicate** - (Optional) One of ByteMatchSet, IPSet, SizeConstraintSet, SqlInjectionMatchSet, or XssMatchSet objects to include in a rule.

## » Nested Blocks

### » predicate

See the WAF Documentation for more information.

## » Arguments

- **negated** - (Required) Set this to **false** if you want to allow, block, or count requests based on the settings in the specified ByteMatchSet, IPSet, SqlInjectionMatchSet, XssMatchSet, or SizeConstraintSet. For example, if an IPSet includes the IP address 192.0.2.44, AWS WAF will allow or block requests based on that IP address. If set to **true**, AWS WAF will allow, block, or count requests based on all IP addresses *except* 192.0.2.44.
- **data\_id** - (Required) A unique identifier for a predicate in the rule, such as Byte Match Set ID or IPSet ID.
- **type** - (Required) The type of predicate in a rule. Valid values: ByteMatch, GeoMatch, IPMatch, RegexMatch, SizeConstraint, SqlInjectionMatch, or XssMatch.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF Regional rate based rule.

## » `aws_wafregional_regex_match_set`

Provides a WAF Regional Regex Match Set Resource

### » Example Usage

```
resource "aws_wafregional_regex_match_set" "example" {
  name = "example"
  regex_match_tuple {
    field_to_match {
      data = "User-Agent"
      type = "HEADER"
    }
    regex_pattern_set_id = "${aws_wafregional_regex_pattern_set.example.id}"
    text_transformation = "NONE"
  }
}

resource "aws_wafregional_regex_pattern_set" "example" {
  name = "example"
  regex_pattern_strings = ["one", "two"]
}
```

### » Argument Reference

The following arguments are supported:

- `name` - (Required) The name or description of the Regex Match Set.
- `regex_match_tuple` - (Required) The regular expression pattern that you want AWS WAF to search for in web requests, the location in requests that you want AWS WAF to search, and other settings. See below.

### » Nested Arguments

#### » `regex_match_tuple`

- `field_to_match` - (Required) The part of a web request that you want to search, such as a specified header or a query string.
- `regex_pattern_set_id` - (Required) The ID of a Regex Pattern Set.
- `text_transformation` - (Required) Text transformations used to eliminate unusual formatting that attackers use in web requests in an effort to bypass AWS WAF. e.g. `CMD_LINE`, `HTML_ENTITY_DECODE` or `NONE`. See docs for all supported values.

#### » `field_to_match`

- **data** - (Optional) When **type** is **HEADER**, enter the name of the header that you want to search, e.g. **User-Agent** or **Referer**. If **type** is any other value, omit this field.
- **type** - (Required) The part of the web request that you want AWS WAF to search for a specified string. e.g. **HEADER**, **METHOD** or **BODY**. See docs for all supported values.

#### » **Attributes Reference**

The following attributes are exported:

- **id** - The ID of the WAF Regional Regex Match Set.

#### » `aws_wafregional_regex_pattern_set`

Provides a WAF Regional Regex Pattern Set Resource

#### » **Example Usage**

```
resource "aws_wafregional_regex_pattern_set" "example" {  
  name = "example"  
  regex_pattern_strings = ["one", "two"]  
}
```

#### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) The name or description of the Regex Pattern Set.
- **regex\_pattern\_strings** - (Optional) A list of regular expression (regex) patterns that you want AWS WAF to search for, such as `B[a@]dB[o0]t`.

#### » **Attributes Reference**

The following attributes are exported:

- **id** - The ID of the WAF Regional Regex Pattern Set.

## » **aws\_\_wafregional\_\_rule**

Provides an WAF Regional Rule Resource for use with Application Load Balancer.

### » **Example Usage**

```
resource "aws_wafregional_ipset" "ipset" {
  name = "tfIPSet"

  ip_set_descriptor {
    type = "IPv4"
    value = "192.0.7.0/24"
  }
}

resource "aws_wafregional_rule" "wafrule" {
  name = "tfWAFRule"
  metric_name = "tfWAFRule"

  predicate {
    type = "IPMatch"
    data_id = "${aws_wafregional_ipset.ipset.id}"
    negated = false
  }
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) The name or description of the rule.
- **metric\_name** - (Required) The name or description for the Amazon CloudWatch metric of this rule.
- **predicate** - (Optional) The objects to include in a rule.

### » **Nested Fields**

#### » **predicate**

See the WAF Documentation for more information.

## » Arguments

- **type** - (Required) The type of predicate in a rule. Valid values: `ByteMatch`, `GeoMatch`, `IPMatch`, `RegexMatch`, `SizeConstraint`, `SqlInjectionMatch`, or `XssMatch`
- **data\_id** - (Required) The unique identifier of a predicate, such as the ID of a `ByteMatchSet` or `IPSet`.
- **negated** - (Required) Whether to use the settings or the negated settings that you specified in the objects.

## » Remarks

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF Regional Rule.

## » `aws_wafregional_rule_group`

Provides a WAF Regional Rule Group Resource

## » Example Usage

```
resource "aws_wafregional_rule" "example" {
  name          = "example"
  metric_name    = "example"
}

resource "aws_wafregional_rule_group" "example" {
  name = "example"
  metric_name = "example"
  activated_rule {
    action {
      type = "COUNT"
    }
  }
  priority = 50
  rule_id = "${aws_wafregional_rule.example.id}"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) A friendly name of the rule group
- **metric\_name** - (Required) A friendly name for the metrics from the rule group
- **activated\_rule** - (Optional) A list of activated rules, see below

## » Nested Blocks

### » **activated\_rule**

#### » Arguments

- **action** - (Required) Specifies the action that CloudFront or AWS WAF takes when a web request matches the conditions in the rule.
  - **type** - (Required) e.g. **BLOCK**, **ALLOW**, or **COUNT**
- **priority** - (Required) Specifies the order in which the rules are evaluated. Rules with a lower value are evaluated before rules with a higher value.
- **rule\_id** - (Required) The ID of a rule
- **type** - (Optional) The rule type, either **REGULAR**, **[RATE\_BASED]**([https://docs/providers/aws/r/wafregional\\_rate\\_based\\_rule\\_group.html](https://docs/providers/aws/r/wafregional_rate_based_rule_group.html)) or **GROUP**. Defaults to **REGULAR**.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF Regional Rule Group.

## » **aws\_wafregional\_size\_constraint\_set**

Provides a WAF Regional Size Constraint Set Resource for use with Application Load Balancer.

## » Example Usage

```
resource "aws_wafregional_size_constraint_set" "size_constraint_set" {
  name = "tfsize_constraints"

  size_constraints {
    text_transformation = "NONE"
    comparison_operator = "EQ"
  }
}
```

```

    size                = "4096"

    field_to_match {
      type = "BODY"
    }
  }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name or description of the Size Constraint Set.
- **size\_constraints** - (Optional) Specifies the parts of web requests that you want to inspect the size of.

## » Nested Blocks

### » `size_constraints`

#### » Arguments

- **field\_to\_match** - (Required) Specifies where in a web request to look for the size constraint.
- **comparison\_operator** - (Required) The type of comparison you want to perform. e.g. EQ, NE, LT, GT. See docs for all supported values.
- **size** - (Required) The size in bytes that you want to compare against the size of the specified **field\_to\_match**. Valid values are between 0 - 21474836480 bytes (0 - 20 GB).
- **text\_transformation** - (Required) Text transformations used to eliminate unusual formatting that attackers use in web requests in an effort to bypass AWS WAF. If you specify a transformation, AWS WAF performs the transformation on **field\_to\_match** before inspecting a request for a match. e.g. CMD\_LINE, HTML\_ENTITY\_DECODE or NONE. See docs for all supported values. **Note:** if you choose BODY as **type**, you must choose NONE because CloudFront forwards only the first 8192 bytes for inspection.

### » `field_to_match`

#### » Arguments

- **data** - (Optional) When **type** is HEADER, enter the name of the header that you want to search, e.g. User-Agent or Referer. If **type** is any other value, omit this field.



- **type** - (Required) The part of the web request that you want AWS WAF to search for a specified string. e.g. `HEADER`, `METHOD` or `BODY`. See docs for all supported values.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF Size Constraint Set.

## » `aws_wafregional_sql_injection_match_set`

Provides a WAF Regional SQL Injection Match Set Resource for use with Application Load Balancer.

## » Example Usage

```
resource "aws_wafregional_sql_injection_match_set" "sql_injection_match_set" {
  name = "tf-sql_injection_match_set"
  sql_injection_match_tuple {
    text_transformation = "URL_DECODE"
    field_to_match {
      type = "QUERY_STRING"
    }
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name or description of the SizeConstraintSet.
- **sql\_injection\_match\_tuple** - (Optional) The parts of web requests that you want AWS WAF to inspect for malicious SQL code and, if you want AWS WAF to inspect a header, the name of the header.

## » Nested fields

### » `sql_injection_match_tuple`

- **field\_to\_match** - (Required) Specifies where in a web request to look for snippets of malicious SQL code.

- **text\_transformation** - (Required) Text transformations used to eliminate unusual formatting that attackers use in web requests in an effort to bypass AWS WAF. If you specify a transformation, AWS WAF performs the transformation on **field\_to\_match** before inspecting a request for a match. e.g. `CMD_LINE`, `HTML_ENTITY_DECODE` or `NONE`. See docs for all supported values.

#### » **field\_to\_match**

- **data** - (Optional) When **type** is `HEADER`, enter the name of the header that you want to search, e.g. `User-Agent` or `Referer`. If **type** is any other value, omit this field.
- **type** - (Required) The part of the web request that you want AWS WAF to search for a specified string. e.g. `HEADER`, `METHOD` or `BODY`. See docs for all supported values.

### » **Attributes Reference**

The following attributes are exported:

- **id** - The ID of the WAF `SqlInjectionMatchSet`.

### » **aws\_wafregional\_web\_acl**

Provides a WAF Regional Web ACL Resource for use with Application Load Balancer.

### » **Example Usage**

```
resource "aws_wafregional_ipset" "ipset" {
  name = "tfIPSet"

  ip_set_descriptor {
    type = "IPV4"
    value = "192.0.7.0/24"
  }
}

resource "aws_wafregional_rule" "wafrule" {
  name          = "tfWAFRule"
  metric_name = "tfWAFRule"
```

```

    predicate {
      data_id = "${aws_wafregional_ipset.ipset.id}"
      negated = false
      type    = "IPMatch"
    }
  }

resource "aws_wafregional_web_acl" "wafacl" {
  name          = "tfWebACL"
  metric_name   = "tfWebACL"

  default_action {
    type = "ALLOW"
  }

  rule {
    action {
      type = "BLOCK"
    }

    priority = 1
    rule_id  = "${aws_wafregional_rule.wafrule.id}"
  }
}

```

## » Argument Reference

The following arguments are supported:

- **default\_action** - (Required) The action that you want AWS WAF Regional to take when a request doesn't match the criteria in any of the rules that are associated with the web ACL.
- **metric\_name** - (Required) The name or description for the Amazon CloudWatch metric of this web ACL.
- **name** - (Required) The name or description of the web ACL.
- **rule** - (Required) The rules to associate with the web ACL and the settings for each rule.

## » Nested Fields

### » rule

See docs for all details and supported values.

#### » Arguments

- **action** - (Required) The action that CloudFront or AWS WAF takes when a web request matches the conditions in the rule.
- **priority** - (Required) Specifies the order in which the rules in a WebACL are evaluated. Rules with a lower value are evaluated before rules with a higher value.
- **rule\_id** - (Required) ID of the associated rule

#### » default\_action / action

#### » Arguments

- **type** - (Required) Specifies how you want AWS WAF Regional to respond to requests that match the settings in a rule. e.g. **ALLOW**, **BLOCK** or **COUNT**

#### » Attributes Reference

The following attributes are exported:

- **id** - The ID of the WAF Regional WebACL.

### » aws\_wafregional\_web\_acl\_association

Provides a resource to create an association between a WAF Regional WebACL and Application Load Balancer.

**Note:** An Application Load Balancer can only be associated with one WAF Regional WebACL.

#### » Example Usage

```
resource "aws_wafregional_ipset" "ipset" {
  name = "tfIPSet"

  ip_set_descriptor {
    type = "IPv4"
    value = "192.0.7.0/24"
  }
}

resource "aws_wafregional_rule" "foo" {
  name          = "tfWAFRule"
  metric_name = "tfWAFRule"
```

```

    predicate {
      data_id = "${aws_wafregional_ipset.ipset.id}"
      negated = false
      type    = "IPMatch"
    }
  }

resource "aws_wafregional_web_acl" "foo" {
  name = "foo"
  metric_name = "foo"
  default_action {
    type = "ALLOW"
  }
  rule {
    action {
      type = "BLOCK"
    }
    priority = 1
    rule_id = "${aws_wafregional_rule.foo.id}"
  }
}

resource "aws_vpc" "foo" {
  cidr_block = "10.1.0.0/16"
}

data "aws_availability_zones" "available" {}

resource "aws_subnet" "foo" {
  vpc_id = "${aws_vpc.foo.id}"
  cidr_block = "10.1.1.0/24"
  availability_zone = "${data.aws_availability_zones.available.names[0]}"
}

resource "aws_subnet" "bar" {
  vpc_id = "${aws_vpc.foo.id}"
  cidr_block = "10.1.2.0/24"
  availability_zone = "${data.aws_availability_zones.available.names[1]}"
}

resource "aws_alb" "foo" {
  internal = true
  subnets = ["${aws_subnet.foo.id}", "${aws_subnet.bar.id}"]
}

```

```
resource "aws_wafregional_web_acl_association" "foo" {
  resource_arn = "${aws_alb.foo.arn}"
  web_acl_id = "${aws_wafregional_web_acl.foo.id}"
}
```

## » Argument Reference

The following arguments are supported:

- **web\_acl\_id** - (Required) The ID of the WAF Regional WebACL to create an association.
- **resource\_arn** - (Required) Application Load Balancer ARN to associate with.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the association

## » aws\_\_wafregional\_\_xss\_\_match\_\_set

Provides a WAF Regional XSS Match Set Resource for use with Application Load Balancer.

## » Example Usage

```
resource "aws_wafregional_xss_match_set" "xss_match_set" {
  name = "xss_match_set"
  xss_match_tuple {
    text_transformation = "NONE"
    field_to_match {
      type = "URI"
    }
  }
}

xss_match_tuple {
  text_transformation = "NONE"
  field_to_match {
    type = "QUERY_STRING"
  }
}
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the set
- **xss\_match\_tuple** - (Optional) The parts of web requests that you want to inspect for cross-site scripting attacks.

### » Nested fields

#### » **xss\_match\_tuple**

- **field\_to\_match** - (Required) Specifies where in a web request to look for cross-site scripting attacks.
- **text\_transformation** - (Required) Which text transformation, if any, to perform on the web request before inspecting the request for cross-site scripting attacks.

#### » **field\_to\_match**

- **data** - (Optional) When the value of **type** is **HEADER**, enter the name of the header that you want the WAF to search, for example, **User-Agent** or **Referer**. If the value of **type** is any other value, omit **data**.
- **type** - (Required) The part of the web request that you want AWS WAF to search for a specified string. e.g. **HEADER** or **METHOD**

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the Regional WAF XSS Match Set.

## » **aws\_route53\_delegation\_set**

Provides a Route53 Delegation Set resource.

### » Example Usage

```
resource "aws_route53_delegation_set" "main" {
  reference_name = "DynDNS"
}

resource "aws_route53_zone" "primary" {
```

```

    name                = "hashicorp.com"
    delegation_set_id = "${aws_route53_delegation_set.main.id}"
}

resource "aws_route53_zone" "secondary" {
    name                = "terraform.io"
    delegation_set_id = "${aws_route53_delegation_set.main.id}"
}

```

## » Argument Reference

The following arguments are supported:

- **reference\_name** - (Optional) This is a reference name used in Caller Reference (helpful for identifying single delegation set amongst others)

## » Attributes Reference

The following attributes are exported:

- **id** - The delegation set ID
- **name\_servers** - A list of authoritative name servers for the hosted zone (effectively a list of NS records).

## » Import

Route53 Delegation Sets can be imported using the `delegation set id`, e.g.

```
$ terraform import aws_route53_delegation_set.set1 N1PA6795SAMPLE
```

## » aws\_\_route53\_\_health\_\_check

Provides a Route53 health check.

## » Example Usage

```

resource "aws_route53_health_check" "child1" {
    fqdn          = "foobar.terraform.com"
    port          = 80
    type          = "HTTP"
    resource_path = "/"
    failure_threshold = "5"
}

```



```

    request_interval = "30"

    tags = {
        Name = "tf-test-health-check"
    }
}

resource "aws_route53_health_check" "foo" {
    type = "CALCULATED"
    child_health_threshold = 1
    child_healthchecks = ["${aws_route53_health_check.child1.id}"]

    tags = {
        Name = "tf-test-calculated-health-check"
    }
}

```

## » CloudWatch Alarm Example

```

resource "aws_cloudwatch_metric_alarm" "foobar" {
    alarm_name          = "terraform-test-foobar5"
    comparison_operator = "GreaterThanOrEqualToThreshold"
    evaluation_periods  = "2"
    metric_name         = "CPUUtilization"
    namespace          = "AWS/EC2"
    period              = "120"
    statistic           = "Average"
    threshold           = "80"
    alarm_description   = "This metric monitors ec2 cpu utilization"
}

resource "aws_route53_health_check" "foo" {
    type = "CLOUDWATCH_METRIC"
    cloudwatch_alarm_name = "${aws_cloudwatch_metric_alarm.foobar.alarm_name}"
    cloudwatch_alarm_region = "us-west-2"
    insufficient_data_health_status = "Healthy"
}

```

## » Argument Reference

The following arguments are supported:

- **reference\_name** - (Optional) This is a reference name used in Caller Reference (helpful for identifying single health\_check set amongst others)

- **fqdn** - (Optional) The fully qualified domain name of the endpoint to be checked.
- **ip\_address** - (Optional) The IP address of the endpoint to be checked.
- **port** - (Optional) The port of the endpoint to be checked.
- **type** - (Required) The protocol to use when performing health checks. Valid values are HTTP, HTTPS, HTTP\_STR\_MATCH, HTTPS\_STR\_MATCH, TCP, CALCULATED and CLOUDWATCH\_METRIC.
- **failure\_threshold** - (Required) The number of consecutive health checks that an endpoint must pass or fail.
- **request\_interval** - (Required) The number of seconds between the time that Amazon Route 53 gets a response from your endpoint and the time that it sends the next health-check request.
- **resource\_path** - (Optional) The path that you want Amazon Route 53 to request when performing health checks.
- **search\_string** - (Optional) String searched in the first 5120 bytes of the response body for check to be considered healthy.
- **measure\_latency** - (Optional) A Boolean value that indicates whether you want Route 53 to measure the latency between health checkers in multiple AWS regions and your endpoint and to display CloudWatch latency graphs in the Route 53 console.
- **invert\_healthcheck** - (Optional) A boolean value that indicates whether the status of health check should be inverted. For example, if a health check is healthy but Inverted is True , then Route 53 considers the health check to be unhealthy.
- **enable\_sni** - (Optional) A boolean value that indicates whether Route53 should send the **fqdn** to the endpoint when performing the health check. This defaults to AWS' defaults: when the **type** is "HTTPS" **enable\_sni** defaults to **true**, when **type** is anything else **enable\_sni** defaults to **false**.
- **child\_healthchecks** - (Optional) For a specified parent health check, a list of HealthCheckId values for the associated child health checks.
- **child\_health\_threshold** - (Optional) The minimum number of child health checks that must be healthy for Route 53 to consider the parent health check to be healthy. Valid values are integers between 0 and 256, inclusive
- **cloudwatch\_alarm\_name** - (Optional) The name of the CloudWatch alarm.
- **cloudwatch\_alarm\_region** - (Optional) The CloudWatchRegion that the CloudWatch alarm was created in.

- `insufficient_data_health_status` - (Optional) The status of the health check when CloudWatch has insufficient data about the state of associated alarm. Valid values are `Healthy` , `Unhealthy` and `LastKnownStatus`.
- `regions` - (Optional) A list of AWS regions that you want Amazon Route 53 health checkers to check the specified endpoint from.
- `tags` - (Optional) A mapping of tags to assign to the health check.

At least one of either `fqdn` or `ip_address` must be specified.

## » Import

Route53 Health Checks can be imported using the `health_check id`, e.g.

```
$ terraform import aws_route53_health_check.http_check abcdef11-2222-3333-4444-555555fedcba
```

## » `aws_route53_query_log`

Provides a Route53 query logging configuration resource.

**NOTE:** There are restrictions on the configuration of query logging. Notably, the CloudWatch log group must be in the `us-east-1` region, a permissive CloudWatch log resource policy must be in place, and the Route53 hosted zone must be public. See [Configuring Logging for DNS Queries](#) for additional details.

## » Example Usage

```
# Example CloudWatch log group in us-east-1

provider "aws" {
  alias   = "us-east-1"
  region = "us-east-1"
}

resource "aws_cloudwatch_log_group" "aws_route53_example_com" {
  provider = "aws.us-east-1"

  name           = "/aws/route53/${aws_route53_zone.example_com.name}"
  retention_in_days = 30
}

# Example CloudWatch log resource policy to allow Route53 to write logs
# to any log group under /aws/route53/*
```

```

data "aws_iam_policy_document" "route53-query-logging-policy" {
  statement {
    actions = [
      "logs:CreateLogStream",
      "logs:PutLogEvents",
    ]

    resources = ["arn:aws:logs:*:*:log-group:/aws/route53/*"]

    principals {
      identifiers = ["route53.amazonaws.com"]
      type       = "Service"
    }
  }
}

resource "aws_cloudwatch_log_resource_policy" "route53-query-logging-policy" {
  provider = "aws.us-east-1"

  policy_document = "${data.aws_iam_policy_document.route53-query-logging-policy.json}"
  policy_name     = "route53-query-logging-policy"
}

# Example Route53 zone with query logging

resource "aws_route53_zone" "example_com" {
  name = "example.com"
}

resource "aws_route53_query_log" "example_com" {
  depends_on = ["aws_cloudwatch_log_resource_policy.route53-query-logging-policy"]

  cloudwatch_log_group_arn = "${aws_cloudwatch_log_group.aws_route53_example_com.arn}"
  zone_id                  = "${aws_route53_zone.example_com.zone_id}"
}

```

## » Argument Reference

The following arguments are supported:

- `cloudwatch_log_group_arn` - (Required) CloudWatch log group ARN to send query logs.
- `zone_id` - (Required) Route53 hosted zone ID to enable query logs.

## » Attributes Reference

The following additional attributes are exported:

- `id` - The query logging configuration ID

## » Import

Route53 query logging configurations can be imported using their ID, e.g.

```
$ terraform import aws_route53_query_log.example_com xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx
```

## » `aws_route53_record`

Provides a Route53 record resource.

## » Example Usage

### » Simple routing policy

```
resource "aws_route53_record" "www" {
  zone_id = "${aws_route53_zone.primary.zone_id}"
  name     = "www.example.com"
  type     = "A"
  ttl      = "300"
  records  = ["${aws_eip.lb.public_ip}"]
}
```

### » Weighted routing policy

Other routing policies are configured similarly. See AWS Route53 Developer Guide for details.

```
resource "aws_route53_record" "www-dev" {
  zone_id = "${aws_route53_zone.primary.zone_id}"
  name     = "www"
  type     = "CNAME"
  ttl      = "5"

  weighted_routing_policy {
    weight = 10
  }
}
```

```

    set_identifier = "dev"
    records        = ["dev.example.com"]
}

resource "aws_route53_record" "www-live" {
  zone_id = "${aws_route53_zone.primary.zone_id}"
  name     = "www"
  type     = "CNAME"
  ttl      = "5"

  weighted_routing_policy {
    weight = 90
  }

  set_identifier = "live"
  records        = ["live.example.com"]
}

```

## » Alias record

See related part of AWS Route53 Developer Guide to understand differences between alias and non-alias records.

TTL for all alias records is 60 seconds, you cannot change this, therefore `ttl` has to be omitted in alias records.

```

resource "aws_elb" "main" {
  name                = "foobar-terraform-elb"
  availability_zones = ["us-east-1c"]

  listener {
    instance_port      = 80
    instance_protocol = "http"
    lb_port            = 80
    lb_protocol        = "http"
  }
}

resource "aws_route53_record" "www" {
  zone_id = "${aws_route53_zone.primary.zone_id}"
  name     = "example.com"
  type     = "A"

  alias {
    name                = "${aws_elb.main.dns_name}"
    zone_id             = "${aws_elb.main.zone_id}"
  }
}

```

```

    evaluate_target_health = true
  }
}

```

## » Argument Reference

The following arguments are supported:

- **zone\_id** - (Required) The ID of the hosted zone to contain this record.
- **name** - (Required) The name of the record.
- **type** - (Required) The record type. Valid values are A, AAAA, CAA, CNAME, MX, NAPTR, NS, PTR, SOA, SPF, SRV and TXT.
- **ttl** - (Required for non-alias records) The TTL of the record.
- **records** - (Required for non-alias records) A string list of records. To specify a single record value longer than 255 characters such as a TXT record for DKIM, add `\\" inside the Terraform configuration string (e.g. "first255characters\\"morecharacters").`
- **set\_identifier** - (Optional) Unique identifier to differentiate records with routing policies from one another. Required if using **failover**, **geolocation**, **latency**, or **weighted** routing policies documented below.
- **health\_check\_id** - (Optional) The health check the record should be associated with.
- **alias** - (Optional) An alias block. Conflicts with **ttl** & **records**. Alias record documented below.
- **failover\_routing\_policy** - (Optional) A block indicating the routing behavior when associated health check fails. Conflicts with any other routing policy. Documented below.
- **geolocation\_routing\_policy** - (Optional) A block indicating a routing policy based on the geolocation of the requestor. Conflicts with any other routing policy. Documented below.
- **latency\_routing\_policy** - (Optional) A block indicating a routing policy based on the latency between the requestor and an AWS region. Conflicts with any other routing policy. Documented below.
- **weighted\_routing\_policy** - (Optional) A block indicating a weighted routing policy. Conflicts with any other routing policy. Documented below.
- **multivalue\_answer\_routing\_policy** - (Optional) A block indicating a multivalue answer routing policy. Conflicts with any other routing policy.
- **allow\_overwrite** - (Optional) Allow creation of this record in Terraform to overwrite an existing record, if any. This does not prevent other resources within Terraform or manual Route53 changes from overwriting this record. **true** by default.

Exactly one of **records** or **alias** must be specified: this determines whether it's an alias record.

Alias records support the following:

- **name** - (Required) DNS domain name for a CloudFront distribution, S3 bucket, ELB, or another resource record set in this hosted zone.
- **zone\_id** - (Required) Hosted zone ID for a CloudFront distribution, S3 bucket, ELB, or Route 53 hosted zone. See **resource\_elb.zone\_id** for example.
- **evaluate\_target\_health** - (Required) Set to **true** if you want Route 53 to determine whether to respond to DNS queries using this resource record set by checking the health of the resource record set. Some resources have special requirements, see related part of documentation.

Failover routing policies support the following:

- **type** - (Required) **PRIMARY** or **SECONDARY**. A **PRIMARY** record will be served if its healthcheck is passing, otherwise the **SECONDARY** will be served. See <http://docs.aws.amazon.com/Route53/latest/DeveloperGuide/dns-failover-configuring-options.html#dns-failover-failover-rrsets>

Geolocation routing policies support the following:

- **continent** - A two-letter continent code. See [http://docs.aws.amazon.com/Route53/latest/APIReference/API\\_GetGeoLocation.html](http://docs.aws.amazon.com/Route53/latest/APIReference/API_GetGeoLocation.html) for code details. Either **continent** or **country** must be specified.
- **country** - A two-character country code or **\*** to indicate a default resource record set.
- **subdivision** - (Optional) A subdivision code for a country.

Latency routing policies support the following:

- **region** - (Required) An AWS region from which to measure latency. See <http://docs.aws.amazon.com/Route53/latest/DeveloperGuide/routing-policy.html#routing-policy-latency>

Weighted routing policies support the following:

- **weight** - (Required) A numeric value indicating the relative weight of the record. See <http://docs.aws.amazon.com/Route53/latest/DeveloperGuide/routing-policy.html#routing-policy-weighted>.

## » Attributes Reference

- **fqdn** - FQDN built using the zone domain and **name**

## » Import

Route53 Records can be imported using ID of the record. The ID is made up as **ZONEID\_RECORDNAME\_TYPE\_SET-IDENTIFIER**



e.g.

```
Z4KAPRWWNC7JR_dev.example.com_NS_dev
```

In this example, Z4KAPRWWNC7JR is the ZoneID, dev.example.com is the Record Name, NS is the Type and dev is the Set Identifier. Only the Set Identifier is actually optional in the ID

To import the ID above, it would look as follows:

```
$ terraform import aws_route53_record.myrecord Z4KAPRWWNC7JR_dev.example.com_NS_dev
```

## » aws\_\_route53\_\_zone

Provides a Route53 Hosted Zone resource.

### » Example Usage

```
resource "aws_route53_zone" "primary" {
  name = "example.com"
}
```

For use in subdomains, note that you need to create a aws\_route53\_record of type NS as well as the subdomain zone.

```
resource "aws_route53_zone" "main" {
  name = "example.com"
}
```

```
resource "aws_route53_zone" "dev" {
  name = "dev.example.com"

  tags {
    Environment = "dev"
  }
}
```

```
resource "aws_route53_record" "dev-ns" {
  zone_id = "${aws_route53_zone.main.zone_id}"
  name    = "dev.example.com"
  type    = "NS"
  ttl     = "30"

  records = [
    "${aws_route53_zone.dev.name_servers.0}",
    "${aws_route53_zone.dev.name_servers.1}",
  ]
}
```

```

    "${aws_route53_zone.dev.name_servers.2}",
    "${aws_route53_zone.dev.name_servers.3}",
  ]
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) This is the name of the hosted zone.
- **comment** - (Optional) A comment for the hosted zone. Defaults to 'Managed by Terraform'.
- **tags** - (Optional) A mapping of tags to assign to the zone.
- **vpc\_id** - (Optional) The VPC to associate with a private hosted zone. Specifying **vpc\_id** will create a private hosted zone. Conflicts w/ **delegation\_set\_id** as delegation sets can only be used for public zones.
- **vpc\_region** - (Optional) The VPC's region. Defaults to the region of the AWS provider.
- **delegation\_set\_id** - (Optional) The ID of the reusable delegation set whose NS records you want to assign to the hosted zone. Conflicts w/ **vpc\_id** as delegation sets can only be used for public zones.
- **force\_destroy** - (Optional) Whether to destroy all records (possibly managed outside of Terraform) in the zone when destroying the zone.

## » Attributes Reference

The following attributes are exported:

- **zone\_id** - The Hosted Zone ID. This can be referenced by zone records.
- **name\_servers** - A list of name servers in associated (or default) delegation set. Find more about delegation sets in AWS docs.

## » Import

Route53 Zones can be imported using the **zone id**, e.g.

```
$ terraform import aws_route53_zone.myzone Z1D633PJN98FT9
```

## » aws\_route53\_zone\_association

Provides a Route53 private Hosted Zone to VPC association resource.

## » Example Usage

```
resource "aws_vpc" "primary" {
  cidr_block      = "10.6.0.0/16"
  enable_dns_hostnames = true
  enable_dns_support = true
}

resource "aws_vpc" "secondary" {
  cidr_block      = "10.7.0.0/16"
  enable_dns_hostnames = true
  enable_dns_support = true
}

resource "aws_route53_zone" "example" {
  name     = "example.com"
  vpc_id   = "${aws_vpc.primary.id}"
}

resource "aws_route53_zone_association" "secondary" {
  zone_id = "${aws_route53_zone.example.zone_id}"
  vpc_id  = "${aws_vpc.secondary.id}"
}
```

## » Argument Reference

The following arguments are supported:

- **zone\_id** - (Required) The private hosted zone to associate.
- **vpc\_id** - (Required) The VPC to associate with the private hosted zone.
- **vpc\_region** - (Optional) The VPC's region. Defaults to the region of the AWS provider.

## » Attributes Reference

The following attributes are exported:

- **id** - The calculated unique identifier for the association.
- **zone\_id** - The ID of the hosted zone for the association.
- **vpc\_id** - The ID of the VPC for the association.
- **vpc\_region** - The region in which the VPC identified by **vpc\_id** was created.

## » `aws_s3_bucket`

Provides a S3 bucket resource.

### » Example Usage

#### » Private Bucket w/ Tags

```
resource "aws_s3_bucket" "b" {
  bucket = "my-tf-test-bucket"
  acl    = "private"

  tags {
    Name           = "My bucket"
    Environment    = "Dev"
  }
}
```

#### » Static Website Hosting

```
resource "aws_s3_bucket" "b" {
  bucket = "s3-website-test.hashicorp.com"
  acl    = "public-read"
  policy = "${file("policy.json")}"

  website {
    index_document = "index.html"
    error_document = "error.html"

    routing_rules = <<EOF
[{"
  "Condition": {
    "KeyPrefixEquals": "docs/"
  },
  "Redirect": {
    "ReplaceKeyPrefixWith": "documents/"
  }
}]
EOF
  }
}
```

## » Using CORS

```
resource "aws_s3_bucket" "b" {
  bucket = "s3-website-test.hashicorp.com"
  acl    = "public-read"

  cors_rule {
    allowed_headers = ["*"]
    allowed_methods = ["PUT", "POST"]
    allowed_origins = ["https://s3-website-test.hashicorp.com"]
    expose_headers  = ["ETag"]
    max_age_seconds = 3000
  }
}
```

## » Using versioning

```
resource "aws_s3_bucket" "b" {
  bucket = "my-tf-test-bucket"
  acl    = "private"

  versioning {
    enabled = true
  }
}
```

## » Enable Logging

```
resource "aws_s3_bucket" "log_bucket" {
  bucket = "my-tf-log-bucket"
  acl    = "log-delivery-write"
}

resource "aws_s3_bucket" "b" {
  bucket = "my-tf-test-bucket"
  acl    = "private"

  logging {
    target_bucket = "${aws_s3_bucket.log_bucket.id}"
    target_prefix = "log/"
  }
}
```

» Using object lifecycle

```
resource "aws_s3_bucket" "bucket" {
  bucket = "my-bucket"
  acl    = "private"

  lifecycle_rule {
    id      = "log"
    enabled = true

    prefix = "log/"
    tags {
      "rule"      = "log"
      "autoclean" = "true"
    }

    transition {
      days          = 15
      storage_class = "ONEZONE_IA"
    }

    transition {
      days          = 30
      storage_class = "STANDARD_IA"
    }

    transition {
      days          = 60
      storage_class = "GLACIER"
    }

    expiration {
      days = 90
    }
  }

  lifecycle_rule {
    id      = "tmp"
    prefix  = "tmp/"
    enabled = true

    expiration {
      date = "2016-01-12"
    }
  }
}
```

```

}

resource "aws_s3_bucket" "versioning_bucket" {
  bucket = "my-versioning-bucket"
  acl    = "private"

  versioning {
    enabled = true
  }

  lifecycle_rule {
    prefix = "config/"
    enabled = true

    noncurrent_version_transition {
      days      = 30
      storage_class = "STANDARD_IA"
    }

    noncurrent_version_transition {
      days      = 60
      storage_class = "GLACIER"
    }

    noncurrent_version_expiration {
      days = 90
    }
  }
}

```

#### » Using replication configuration

```

provider "aws" {
  region = "eu-west-1"
}

provider "aws" {
  alias   = "central"
  region = "eu-central-1"
}

resource "aws_iam_role" "replication" {
  name = "tf-iam-role-replication-12345"

  assume_role_policy = <<POLICY

```

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "s3.amazonaws.com"
      },
      "Effect": "Allow",
      "Sid": ""
    }
  ]
}
POLICY
}

resource "aws_iam_policy" "replication" {
  name = "tf-iam-role-policy-replication-12345"

  policy = <<POLICY
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetReplicationConfiguration",
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": [
        "${aws_s3_bucket.bucket.arn}"
      ]
    },
    {
      "Action": [
        "s3:GetObjectVersion",
        "s3:GetObjectVersionAcl"
      ],
      "Effect": "Allow",
      "Resource": [
        "${aws_s3_bucket.bucket.arn}/*"
      ]
    },
    {
      "Action": [
        "s3:ReplicateObject",

```



```

        "s3:ReplicateDelete"
    ],
    "Effect": "Allow",
    "Resource": "${aws_s3_bucket.destination.arn}/*"
}
]
}
POLICY
}

resource "aws_iam_policy_attachment" "replication" {
    name      = "tf-iam-role-attachment-replication-12345"
    roles     = ["${aws_iam_role.replication.name}"]
    policy_arn = "${aws_iam_policy.replication.arn}"
}

resource "aws_s3_bucket" "destination" {
    bucket     = "tf-test-bucket-destination-12345"
    region     = "eu-west-1"

    versioning {
        enabled = true
    }
}

resource "aws_s3_bucket" "bucket" {
    provider = "aws.central"
    bucket   = "tf-test-bucket-12345"
    acl      = "private"
    region   = "eu-central-1"

    versioning {
        enabled = true
    }

    replication_configuration {
        role = "${aws_iam_role.replication.arn}"

        rules {
            id      = "foobar"
            prefix  = "foo"
            status  = "Enabled"

            destination {
                bucket     = "${aws_s3_bucket.destination.arn}"
                storage_class = "STANDARD"
            }
        }
    }
}

```

```

    }
  }
}

```

## » Enable Default Server Side Encryption

```

resource "aws_kms_key" "mykey" {
  description          = "This key is used to encrypt bucket objects"
  deletion_window_in_days = 10
}

resource "aws_s3_bucket" "mybucket" {
  bucket = "mybucket"
  server_side_encryption_configuration {
    rule {
      apply_server_side_encryption_by_default {
        kms_master_key_id = "${aws_kms_key.mykey.arn}"
        sse_algorithm      = "aws:kms"
      }
    }
  }
}

```

## » Argument Reference

The following arguments are supported:

- **bucket** - (Optional, Forces new resource) The name of the bucket. If omitted, Terraform will assign a random, unique name.
- **bucket\_prefix** - (Optional, Forces new resource) Creates a unique bucket name beginning with the specified prefix. Conflicts with **bucket**.
- **acl** - (Optional) The canned ACL to apply. Defaults to "private".
- **policy** - (Optional) A valid bucket policy JSON document. Note that if the policy document is not specific enough (but still valid), Terraform may view the policy as constantly changing in a **terraform plan**. In this case, please make sure you use the verbose/specific version of the policy.
- **tags** - (Optional) A mapping of tags to assign to the bucket.
- **force\_destroy** - (Optional, Default:false ) A boolean that indicates all objects should be deleted from the bucket so that the bucket can be destroyed without error. These objects are *not* recoverable.
- **website** - (Optional) A website object (documented below).

- **cors\_rule** - (Optional) A rule of Cross-Origin Resource Sharing (documented below).
- **versioning** - (Optional) A state of versioning (documented below)
- **logging** - (Optional) A settings of bucket logging (documented below).
- **lifecycle\_rule** - (Optional) A configuration of object lifecycle management (documented below).
- **acceleration\_status** - (Optional) Sets the accelerate configuration of an existing bucket. Can be **Enabled** or **Suspended**.
- **region** - (Optional) If specified, the AWS region this bucket should reside in. Otherwise, the region used by the callee.
- **request\_payer** - (Optional) Specifies who should bear the cost of Amazon S3 data transfer. Can be either **BucketOwner** or **Requester**. By default, the owner of the S3 bucket would incur the costs of any data transfer. See Requester Pays Buckets developer guide for more information.
- **replication\_configuration** - (Optional) A configuration of replication configuration (documented below).
- **server\_side\_encryption\_configuration** - (Optional) A configuration of server-side encryption configuration (documented below)

**NOTE:** You cannot use **acceleration\_status** in **cn-north-1** or **us-gov-west-1**

The **website** object supports the following:

- **index\_document** - (Required, unless using **redirect\_all\_requests\_to**) Amazon S3 returns this index document when requests are made to the root domain or any of the subfolders.
- **error\_document** - (Optional) An absolute path to the document to return in case of a 4XX error.
- **redirect\_all\_requests\_to** - (Optional) A hostname to redirect all website requests for this bucket to. Hostname can optionally be prefixed with a protocol (**http://** or **https://**) to use when redirecting requests. The default is the protocol that is used in the original request.
- **routing\_rules** - (Optional) A json array containing routing rules describing redirect behavior and when redirects are applied.

The **CORS** object supports the following:

- **allowed\_headers** (Optional) Specifies which headers are allowed.
- **allowed\_methods** (Required) Specifies which methods are allowed. Can be **GET**, **PUT**, **POST**, **DELETE** or **HEAD**.
- **allowed\_origins** (Required) Specifies which origins are allowed.
- **expose\_headers** (Optional) Specifies expose header in the response.
- **max\_age\_seconds** (Optional) Specifies time in seconds that browser can cache the response for a preflight request.

The **versioning** object supports the following:

- **enabled** - (Optional) Enable versioning. Once you version-enable a bucket, it can never return to an unversioned state. You can, however, suspend versioning on that bucket.
- **mfa\_delete** - (Optional) Enable MFA delete for either **Change the versioning state of your bucket** or **Permanently delete an object version**. Default is **false**.

The **logging** object supports the following:

- **target\_bucket** - (Required) The name of the bucket that will receive the log objects.
- **target\_prefix** - (Optional) To specify a key prefix for log objects.

The **lifecycle\_rule** object supports the following:

- **id** - (Optional) Unique identifier for the rule.
- **prefix** - (Optional) Object key prefix identifying one or more objects to which the rule applies.
- **tags** - (Optional) Specifies object tags key and value.
- **enabled** - (Required) Specifies lifecycle rule status.
- **abort\_incomplete\_multipart\_upload\_days** (Optional) Specifies the number of days after initiating a multipart upload when the multipart upload must be completed.
- **expiration** - (Optional) Specifies a period in the object's expire (documented below).
- **transition** - (Optional) Specifies a period in the object's transitions (documented below).
- **noncurrent\_version\_expiration** - (Optional) Specifies when noncurrent object versions expire (documented below).
- **noncurrent\_version\_transition** - (Optional) Specifies when noncurrent object versions transitions (documented below).

At least one of **expiration**, **transition**, **noncurrent\_version\_expiration**, **noncurrent\_version\_transition** must be specified.

The **expiration** object supports the following

- **date** (Optional) Specifies the date after which you want the corresponding action to take effect.
- **days** (Optional) Specifies the number of days after object creation when the specific rule action takes effect.
- **expired\_object\_delete\_marker** (Optional) On a versioned bucket (versioning-enabled or versioning-suspended bucket), you can add this element in the lifecycle configuration to direct Amazon S3 to delete expired object delete markers.

The **transition** object supports the following

- **date** (Optional) Specifies the date after which you want the corresponding action to take effect.
- **days** (Optional) Specifies the number of days after object creation when the specific rule action takes effect.
- **storage\_class** (Required) Specifies the Amazon S3 storage class to which you want the object to transition. Can be **ONEZONE\_IA**, **STANDARD\_IA**, or **GLACIER**.

The **noncurrent\_version\_expiration** object supports the following

- **days** (Required) Specifies the number of days an object is noncurrent object versions expire.

The **noncurrent\_version\_transition** object supports the following

- **days** (Required) Specifies the number of days an object is noncurrent object versions expire.
- **storage\_class** (Required) Specifies the Amazon S3 storage class to which you want the noncurrent versions object to transition. Can be **ONEZONE\_IA**, **STANDARD\_IA**, or **GLACIER**.

The **replication\_configuration** object supports the following:

- **role** - (Required) The ARN of the IAM role for Amazon S3 to assume when replicating the objects.
- **rules** - (Required) Specifies the rules managing the replication (documented below).

The **rules** object supports the following:

- **id** - (Optional) Unique identifier for the rule.
- **destination** - (Required) Specifies the destination for the rule (documented below).
- **source\_selection\_criteria** - (Optional) Specifies special object selection criteria (documented below).
- **prefix** - (Required) Object keyname prefix identifying one or more objects to which the rule applies. Set as an empty string to replicate the whole bucket.
- **status** - (Required) The status of the rule. Either **Enabled** or **Disabled**. The rule is ignored if status is not **Enabled**.

The **destination** object supports the following:

- **bucket** - (Required) The ARN of the S3 bucket where you want Amazon S3 to store replicas of the object identified by the rule.
- **storage\_class** - (Optional) The class of storage used to store the object.
- **replica\_kms\_key\_id** - (Optional) Destination KMS encryption key ID for SSE-KMS replication. Must be used in conjunction with **sse\_kms\_encrypted\_objects** source selection criteria.

The **source\_selection\_criteria** object supports the following:

- **sse\_kms\_encrypted\_objects** - (Optional) Match SSE-KMS encrypted objects (documented below). If specified, **replica\_kms\_key\_id** in **destination** must be specified as well.

The **sse\_kms\_encrypted\_objects** object supports the following:

- **enabled** - (Required) Boolean which indicates if this criteria is enabled.

The **server\_side\_encryption\_configuration** object supports the following:

- **rule** - (required) A single object for server-side encryption by default configuration. (documented below)

The **rule** object supports the following:

- **apply\_server\_side\_encryption\_by\_default** - (required) A single object for setting server-side encryption by default. (documented below)

The **apply\_server\_side\_encryption\_by\_default** object supports the following:

- **sse\_algorithm** - (required) The server-side encryption algorithm to use. Valid values are **AES256** and **aws:kms**
- **kms\_master\_key\_id** - (optional) The AWS KMS master key ID used for the SSE-KMS encryption. This can only be used when you set the value of **sse\_algorithm** as **aws:kms**. The default **aws/s3** AWS KMS master key is used if this element is absent while the **sse\_algorithm** is **aws:kms**.

## » Attributes Reference

The following attributes are exported:

- **id** - The name of the bucket.
- **arn** - The ARN of the bucket. Will be of format **arn:aws:s3:::bucketname**.
- **bucket\_domain\_name** - The bucket domain name. Will be of format **bucketname.s3.amazonaws.com**.
- **hosted\_zone\_id** - The Route 53 Hosted Zone ID for this bucket's region.
- **region** - The AWS region this bucket resides in.
- **website\_endpoint** - The website endpoint, if the bucket is configured with a website. If not, this will be an empty string.
- **website\_domain** - The domain of the website endpoint, if the bucket is configured with a website. If not, this will be an empty string. This is used to create Route 53 alias records.

## » Import

S3 bucket can be imported using the **bucket**, e.g.

```
$ terraform import aws_s3_bucket.bucket bucket-name
```

## » `aws_s3_bucket_metric`

Provides a S3 bucket metrics configuration resource.

### » Example Usage

#### » Add metrics configuration for entire S3 bucket

```
resource "aws_s3_bucket" "example" {
  bucket = "example"
}

resource "aws_s3_bucket_metric" "example-entire-bucket" {
  bucket = "${aws_s3_bucket.example.bucket}"
  name   = "EntireBucket"
}
```

#### » Add metrics configuration with S3 bucket object filter

```
resource "aws_s3_bucket" "example" {
  bucket = "example"
}

resource "aws_s3_bucket_metric" "example-filtered" {
  bucket = "${aws_s3_bucket.example.bucket}"
  name   = "ImportantBlueDocuments"

  filter {
    prefix = "documents/"

    tags {
      priority = "high"
      class   = "blue"
    }
  }
}
```

### » Argument Reference

The following arguments are supported:

- `bucket` - (Required) The name of the bucket to put metric configuration.

- **name** - (Required) Unique identifier of the metrics configuration for the bucket.
- **filter** - (Optional) Object filtering that accepts a prefix, tags, or a logical AND of prefix and tags (documented below).

The **filter** metric configuration supports the following:

- **prefix** - (Optional) Object prefix for filtering (singular).
- **tags** - (Optional) Object tags for filtering (up to 10).

## » Import

S3 bucket metric configurations can be imported using `bucket:metric`, e.g.

```
$ terraform import aws_s3_bucket_metric.my-bucket-entire-bucket my-bucket:EntireBucket
```

## » `aws_s3_bucket_notification`

Provides a S3 bucket notification resource.

## » Example Usage

### » Add notification configuration to SNS Topic

```
resource "aws_sns_topic" "topic" {
  name = "s3-event-notification-topic"

  policy = <<POLICY
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {"AWS": "*"},
    "Action": "SNS:Publish",
    "Resource": "arn:aws:sns:*:*:s3-event-notification-topic",
    "Condition": {
      "ArnLike": {"aws:SourceArn": "${aws_s3_bucket.bucket.arn}"}}
  }]
}
POLICY
}
```

```
resource "aws_s3_bucket" "bucket" {
```



```

    bucket = "your_bucket_name"
  }

  resource "aws_s3_bucket_notification" "bucket_notification" {
    bucket = "${aws_s3_bucket.bucket.id}"

    topic {
      topic_arn      = "${aws_sns_topic.topic.arn}"
      events          = ["s3:ObjectCreated:*"]
      filter_suffix = ".log"
    }
  }
}

```

#### » Add notification configuration to SQS Queue

```

resource "aws_sqs_queue" "queue" {
  name = "s3-event-notification-queue"

  policy = <<POLICY
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "sqs:SendMessage",
      "Resource": "arn:aws:sqs:*:*:s3-event-notification-queue",
      "Condition": {
        "ArnEquals": { "aws:SourceArn": "${aws_s3_bucket.bucket.arn}" }
      }
    }
  ]
}
POLICY
}

resource "aws_s3_bucket" "bucket" {
  bucket = "your_bucket_name"
}

resource "aws_s3_bucket_notification" "bucket_notification" {
  bucket = "${aws_s3_bucket.bucket.id}"

  queue {
    queue_arn      = "${aws_sqs_queue.queue.arn}"

```

```

        events          = ["s3:ObjectCreated:*"]
        filter_suffix = ".log"
    }
}

```

## » Add notification configuration to Lambda Function

```

resource "aws_iam_role" "iam_for_lambda" {
    name = "iam_for_lambda"

    assume_role_policy = <<EOF
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "sts:AssumeRole",
            "Principal": {
                "Service": "lambda.amazonaws.com"
            },
            "Effect": "Allow"
        }
    ]
}
    EOF
}

resource "aws_lambda_permission" "allow_bucket" {
    statement_id = "AllowExecutionFromS3Bucket"
    action       = "lambda:InvokeFunction"
    function_name = "${aws_lambda_function.func.arn}"
    principal     = "s3.amazonaws.com"
    source_arn     = "${aws_s3_bucket.bucket.arn}"
}

resource "aws_lambda_function" "func" {
    filename      = "your-function.zip"
    function_name = "example_lambda_name"
    role          = "${aws_iam_role.iam_for_lambda.arn}"
    handler       = "exports.example"
}

resource "aws_s3_bucket" "bucket" {
    bucket = "your_bucket_name"
}

```

```

resource "aws_s3_bucket_notification" "bucket_notification" {
  bucket = "${aws_s3_bucket.bucket.id}"

  lambda_function {
    lambda_function_arn = "${aws_lambda_function.func.arn}"
    events               = ["s3:ObjectCreated:*"]
    filter_prefix        = "AWSLogs/"
    filter_suffix        = ".log"
  }
}

```

## » Trigger multiple Lambda functions

```

resource "aws_iam_role" "iam_for_lambda" {
  name = "iam_for_lambda"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Principal": {
        "Service": "lambda.amazonaws.com"
      },
      "Effect": "Allow"
    }
  ]
}
EOF
}

resource "aws_lambda_permission" "allow_bucket1" {
  statement_id = "AllowExecutionFromS3Bucket1"
  action       = "lambda:InvokeFunction"
  function_name = "${aws_lambda_function.func1.arn}"
  principal     = "s3.amazonaws.com"
  source_arn    = "${aws_s3_bucket.bucket.arn}"
}

resource "aws_lambda_function" "func1" {
  filename      = "your-function1.zip"
  function_name = "example_lambda_name1"
  role          = "${aws_iam_role.iam_for_lambda.arn}"
  handler       = "exports.example"
}

```

```

}

resource "aws_lambda_permission" "allow_bucket2" {
  statement_id = "AllowExecutionFromS3Bucket2"
  action       = "lambda:InvokeFunction"
  function_name = "${aws_lambda_function.func2.arn}"
  principal    = "s3.amazonaws.com"
  source_arn    = "${aws_s3_bucket.bucket.arn}"
}

resource "aws_lambda_function" "func2" {
  filename      = "your-function2.zip"
  function_name = "example_lambda_name2"
  role          = "${aws_iam_role.iam_for_lambda.arn}"
  handler       = "exports.example"
}

resource "aws_s3_bucket" "bucket" {
  bucket = "your_bucket_name"
}

resource "aws_s3_bucket_notification" "bucket_notification" {
  bucket = "${aws_s3_bucket.bucket.id}"

  lambda_function {
    lambda_function_arn = "${aws_lambda_function.func1.arn}"
    events               = ["s3:ObjectCreated:*"]
    filter_prefix        = "AWSLogs/"
    filter_suffix        = ".log"
  }

  lambda_function {
    lambda_function_arn = "${aws_lambda_function.func2.arn}"
    events               = ["s3:ObjectCreated:*"]
    filter_prefix        = "OtherLogs/"
    filter_suffix        = ".log"
  }
}

```

## » Add multiple notification configurations to SQS Queue

```

resource "aws_sqs_queue" "queue" {
  name = "s3-event-notification-queue"

  policy = <<POLICY

```

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "sqs:SendMessage",
      "Resource": "arn:aws:sqs:*:*:s3-event-notification-queue",
      "Condition": {
        "ArnEquals": { "aws:SourceArn": "${aws_s3_bucket.bucket.arn}" }
      }
    }
  ]
}
POLICY
}

resource "aws_s3_bucket" "bucket" {
  bucket = "your_bucket_name"
}

resource "aws_s3_bucket_notification" "bucket_notification" {
  bucket = "${aws_s3_bucket.bucket.id}"

  queue {
    id           = "image-upload-event"
    queue_arn    = "${aws_sqs_queue.queue.arn}"
    events       = ["s3:ObjectCreated:*"]
    filter_prefix = "images/"
  }

  queue {
    id           = "video-upload-event"
    queue_arn    = "${aws_sqs_queue.queue.arn}"
    events       = ["s3:ObjectCreated:*"]
    filter_prefix = "videos/"
  }
}

```

For Terraform's JSON syntax, use an array instead of defining the `queue` key twice.

```

{
  "bucket": "${aws_s3_bucket.bucket.id}",
  "queue": [
    {
      "id": "image-upload-event",

```

```

        "queue_arn": "${aws_sqs_queue.queue.arn}",
        "events": ["s3:ObjectCreated:*"],
        "filter_prefix": "images/"
    },
    {
        "id": "video-upload-event",
        "queue_arn": "${aws_sqs_queue.queue.arn}",
        "events": ["s3:ObjectCreated:*"],
        "filter_prefix": "videos/"
    }
]
}

```

## » Argument Reference

The following arguments are supported:

- **bucket** - (Required) The name of the bucket to put notification configuration.
- **topic** - (Optional) The notification configuration to SNS Topic (documented below).
- **queue** - (Optional) The notification configuration to SQS Queue (documented below).
- **lambda\_function** - (Optional, Multiple) Used to configure notifications to a Lambda Function (documented below).

The **topic** notification configuration supports the following:

- **id** - (Optional) Specifies unique identifier for each of the notification configurations.
- **topic\_arn** - (Required) Specifies Amazon SNS topic ARN.
- **events** - (Required) Specifies event for which to send notifications.
- **filter\_prefix** - (Optional) Specifies object key name prefix.
- **filter\_suffix** - (Optional) Specifies object key name suffix.

The **queue** notification configuration supports the following:

- **id** - (Optional) Specifies unique identifier for each of the notification configurations.
- **queue\_arn** - (Required) Specifies Amazon SQS queue ARN.
- **events** - (Required) Specifies event for which to send notifications.
- **filter\_prefix** - (Optional) Specifies object key name prefix.
- **filter\_suffix** - (Optional) Specifies object key name suffix.

The **lambda\_function** notification configuration supports the following:

- **id** - (Optional) Specifies unique identifier for each of the notification configurations.

- `lambda_function_arn` - (Required) Specifies Amazon Lambda function ARN.
- `events` - (Required) Specifies event for which to send notifications.
- `filter_prefix` - (Optional) Specifies object key name prefix.
- `filter_suffix` - (Optional) Specifies object key name suffix.

## » Import

S3 bucket notification can be imported using the `bucket`, e.g.

```
$ terraform import aws_s3_bucket_notification.bucket_notification bucket-name
```

## » `aws_s3_bucket_object`

Provides a S3 bucket object resource.

## » Example Usage

### » Uploading a file to a bucket

```
resource "aws_s3_bucket_object" "object" {
  bucket = "your_bucket_name"
  key    = "new_object_key"
  source = "path/to/file"
  etag   = "${md5(file("path/to/file"))}"
}
```

### » Encrypting with KMS Key

```
resource "aws_kms_key" "examplekms" {
  description          = "KMS key 1"
  deletion_window_in_days = 7
}
```

```
resource "aws_s3_bucket" "examplebucket" {
  bucket = "examplebucketttftest"
  acl    = "private"
}
```

```
resource "aws_s3_bucket_object" "examplebucket_object" {
  key      = "someobject"
  bucket   = "${aws_s3_bucket.examplebucket.id}"
}
```

```

    source      = "index.html"
    kms_key_id = "${aws_kms_key.examplekms.arn}"
}

```

## » Server Side Encryption with S3 Default Master Key

```

resource "aws_s3_bucket" "examplebucket" {
    bucket = "examplebucketttftest"
    acl    = "private"
}

resource "aws_s3_bucket_object" "examplebucket_object" {
    key                = "someobject"
    bucket             = "${aws_s3_bucket.examplebucket.id}"
    source             = "index.html"
    server_side_encryption = "aws:kms"
}

```

## » Argument Reference

**Note:** If you specify `content_encoding` you are responsible for encoding the body appropriately. `source`, `content`, and `content_base64` all expect already encoded/compressed bytes.

The following arguments are supported:

- **bucket** - (Required) The name of the bucket to put the file in.
- **key** - (Required) The name of the object once it is in the bucket.
- **source** - (Required unless `content` or `content_base64` is set) The path to a file that will be read and uploaded as raw bytes for the object content.
- **content** - (Required unless `source` or `content_base64` is set) Literal string value to use as the object content, which will be uploaded as UTF-8-encoded text.
- **content\_base64** - (Required unless `source` or `content` is set) Base64-encoded data that will be decoded and uploaded as raw bytes for the object content. This allows safely uploading non-UTF8 binary data, but is recommended only for small content such as the result of the `gzipbase64` function with small text strings. For larger objects, use `source` to stream the content from a disk file.
- **acl** - (Optional) The canned ACL to apply. Defaults to "private".
- **cache\_control** - (Optional) Specifies caching behavior along the request/reply chain Read w3c `cache_control` for further details.
- **content\_disposition** - (Optional) Specifies presentational information for the object. Read w3c `content_disposition` for further information.



- **content\_encoding** - (Optional) Specifies what content encodings have been applied to the object and thus what decoding mechanisms must be applied to obtain the media-type referenced by the Content-Type header field. Read w3c content encoding for further information.
- **content\_language** - (Optional) The language the content is in e.g. en-US or en-GB.
- **content\_type** - (Optional) A standard MIME type describing the format of the object data, e.g. application/octet-stream. All Valid MIME Types are valid for this input.
- **website\_redirect** - (Optional) Specifies a target URL for website redirect.
- **storage\_class** - (Optional) Specifies the desired Storage Class for the object. Can be either "STANDARD", "REDUCED\_REDUNDANCY", "ONEZONE\_IA", or "STANDARD\_IA". Defaults to "STANDARD".
- **etag** - (Optional) Used to trigger updates. The only meaningful value is `${md5(file("path/to/file"))}`. This attribute is not compatible with **kms\_key\_id**.
- **server\_side\_encryption** - (Optional) Specifies server-side encryption of the object in S3. Valid values are "AES256" and "aws:kms".
- **kms\_key\_id** - (Optional) Specifies the AWS KMS Key ARN to use for object encryption. This value is a fully qualified **ARN** of the KMS Key. If using **aws\_kms\_key**, use the exported **arn** attribute: **kms\_key\_id** = `"${aws_kms_key.foo.arn}"`
- **tags** - (Optional) A mapping of tags to assign to the object.

Either **source** or **content** must be provided to specify the bucket content. These two arguments are mutually-exclusive.

## » Attributes Reference

The following attributes are exported

- **id** - the **key** of the resource supplied above
- **etag** - the ETag generated for the object (an MD5 sum of the object content).
- **version\_id** - A unique version ID value for the object, if bucket versioning is enabled.

## » **aws\_s3\_bucket\_policy**

Attaches a policy to an S3 bucket resource.

## » Example Usage

### » Basic Usage

```
resource "aws_s3_bucket" "b" {
  bucket = "my_tf_test_bucket"
}

resource "aws_s3_bucket_policy" "b" {
  bucket = "${aws_s3_bucket.b.id}"
  policy = <<POLICY
{
  "Version": "2012-10-17",
  "Id": "MYBUCKETPOLICY",
  "Statement": [
    {
      "Sid": "IPAllow",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": "arn:aws:s3::my_tf_test_bucket/*",
      "Condition": {
        "IpAddress": {"aws:SourceIp": "8.8.8.8/32"}
      }
    }
  ]
}
POLICY
}
```

### » Argument Reference

The following arguments are supported:

- `bucket` - (Required) The name of the bucket to which to apply the policy.
- `policy` - (Required) The text of the policy.

### » `aws__secretsmanager__secret`

Provides a resource to manage AWS Secrets Manager secret metadata. To manage a secret value, see the `aws_secretsmanager_secret_version` resource.

## » Example Usage

### » Basic

```
resource "aws_secretsmanager_secret" "example" {  
  name = "example"  
}
```

### » Rotation Configuration

To enable automatic secret rotation, the Secrets Manager service requires usage of a Lambda function. The Rotate Secrets section in the Secrets Manager User Guide provides additional information about deploying a prebuilt Lambda functions for supported credential rotation (e.g. RDS) or deploying a custom Lambda function.

**NOTE:** Configuring rotation causes the secret to rotate once as soon as you store the secret. Before you do this, you must ensure that all of your applications that use the credentials stored in the secret are updated to retrieve the secret from AWS Secrets Manager. The old credentials might no longer be usable after the initial rotation and any applications that you fail to update will break as soon as the old credentials are no longer valid.

**NOTE:** If you cancel a rotation that is in progress (by removing the `rotation` configuration), it can leave the `VersionStage` labels in an unexpected state. Depending on what step of the rotation was in progress, you might need to remove the staging label `AWSPENDING` from the partially created version, specified by the `SecretVersionId` response value. You should also evaluate the partially rotated new version to see if it should be deleted, which you can do by removing all staging labels from the new version's `VersionStage` field.

```
resource "aws_secretsmanager_secret" "rotation-example" {  
  name                = "rotation-example"  
  rotation_lambda_arn = "${aws_lambda_function.example.arn}"  
  
  rotation_rules {  
    automatically_after_days = 7  
  }  
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) Specifies the friendly name of the new secret. The secret name can consist of uppercase letters, lowercase letters, digits, and any of the following characters: `/_+=.@-`. Spaces are not permitted.
- **description** - (Optional) A description of the secret.
- **kms\_key\_id** - (Optional) Specifies the ARN or alias of the AWS KMS customer master key (CMK) to be used to encrypt the secret values in the versions stored in this secret. If you don't specify this value, then Secrets Manager defaults to using the AWS account's default CMK (the one named `aws/secretsmanager`). If the default KMS CMK with that name doesn't yet exist, then AWS Secrets Manager creates it for you automatically the first time.
- **recovery\_window\_in\_days** - (Optional) Specifies the number of days that AWS Secrets Manager waits before it can delete the secret. This value can range from 7 to 30 days. The default value is 30.
- **rotation\_lambda\_arn** - (Optional) Specifies the ARN of the Lambda function that can rotate the secret.
- **rotation\_rules** - (Optional) A structure that defines the rotation configuration for this secret. Defined below.
- **tags** - (Optional) Specifies a key-value map of user-defined tags that are attached to the secret.

#### » **rotation\_rules**

- **automatically\_after\_days** - (Required) Specifies the number of days between automatic scheduled rotations of the secret.

#### » **Attribute Reference**

- **id** - Amazon Resource Name (ARN) of the secret.
- **arn** - Amazon Resource Name (ARN) of the secret.
- **rotation\_enabled** - Specifies whether automatic rotation is enabled for this secret.

#### » **Import**

`aws_secretsmanager_secret` can be imported by using the secret Amazon Resource Name (ARN), e.g.

```
$ terraform import aws_secretsmanager_secret.example arn:aws:secretsmanager:us-east-1:123456
```

## » `aws_secretsmanager_secret_version`

Provides a resource to manage AWS Secrets Manager secret version including its secret value. To manage secret metadata, see the `aws_secretsmanager_secret` resource.

**NOTE:** If the `AWSCURRENT` staging label is present on this version during resource deletion, that label cannot be removed and will be skipped to prevent errors when fully deleting the secret. That label will leave this secret version active even after the resource is deleted from Terraform unless the secret itself is deleted. Move the `AWSCURRENT` staging label before or after deleting this resource from Terraform to fully trigger version deprecation if necessary.

### » Example Usage

```
resource "aws_secretsmanager_secret_version" "example" {
  secret_id      = "${aws_secretsmanager_secret.example.id}"
  secret_string = "example-string-to-protect"
}
```

### » Argument Reference

The following arguments are supported:

- `secret_id` - (Required) Specifies the secret to which you want to add a new version. You can specify either the Amazon Resource Name (ARN) or the friendly name of the secret. The secret must already exist.
- `secret_string` - (Required) Specifies text data that you want to encrypt and store in this version of the secret.
- `version_stages` - (Optional) Specifies a list of staging labels that are attached to this version of the secret. A staging label must be unique to a single version of the secret. If you specify a staging label that's already associated with a different version of the same secret then that staging label is automatically removed from the other version and attached to this version. If you do not specify a value, then AWS Secrets Manager automatically moves the staging label `AWSCURRENT` to this new version on creation.

**NOTE:** If `version_stages` is configured, you must include the `AWSCURRENT` staging label if this secret version is the only version or if the label is currently present on this secret version, otherwise Terraform will show a perpetual difference.

## » Attribute Reference

- **id** - A pipe delimited combination of secret ID and version ID
- **version\_id** - The unique identifier of the version of the secret.

## » Import

`aws_secretsmanager_secret_version` can be imported by using the secret ID and version ID, e.g.

```
$ terraform import aws_secretsmanager_secret.example arn:aws:secretsmanager:us-east-1:123456
```

```
» aws_ses_active_receipt_rule_set
```

Provides a resource to designate the active SES receipt rule set

## » Example Usage

```
resource "aws_ses_active_receipt_rule_set" "main" {
  rule_set_name = "primary-rules"
}
```

## » Argument Reference

The following arguments are supported:

- **rule\_set\_name** - (Required) The name of the rule set

```
» aws__ses__domain__identity
```

Provides an SES domain identity resource

## » Argument Reference

The following arguments are supported:

- **domain** - (Required) The domain name assign to SES

## » Attributes Reference

The following attributes are exported:

- **arn** - The ARN of the domain identity.
- **verification\_token** - A code which when added to the domain as a TXT record will signal to SES that the owner of the domain has authorised SES to act on their behalf. The domain identity will be in state "verification pending" until this is done. See below for an example of how this might be achieved when the domain is hosted in Route 53 and managed by Terraform. Find out more about verifying domains in Amazon SES in the AWS SES docs.

## » Example Usage

```
resource "aws_ses_domain_identity" "example" {
  domain = "example.com"
}

resource "aws_route53_record" "example_amazonses_verification_record" {
  zone_id = "ABCDEFGHIJ123"
  name     = "_amazonses.example.com"
  type     = "TXT"
  ttl      = "600"
  records  = ["${aws_ses_domain_identity.example.verification_token}"]
}
```

## » aws\_\_ses\_\_domain\_\_identity\_\_verification

Represents a successful verification of an SES domain identity.

Most commonly, this resource is used together with `aws_route53_record` and `aws_ses_domain_identity` to request an SES domain identity, deploy the required DNS verification records, and wait for verification to complete.

**WARNING:** This resource implements a part of the verification workflow. It does not represent a real-world entity in AWS, therefore changing or deleting this resource on its own has no immediate effect.

## » Example Usage

```
resource "aws_ses_domain_identity" "example" {
  domain = "example.com"
}
```

```

}

resource "aws_route53_record" "example_amazonses_verification_record" {
  zone_id = "${aws_route53_zone.example.id}"
  name     = "_amazonses.${aws_ses_domain_identity.example.id}"
  type     = "TXT"
  ttl      = "600"
  records  = ["${aws_ses_domain_identity.example.verification_token}"]
}

resource "aws_ses_domain_identity_verification" "example_verification" {
  domain = "${aws_ses_domain_identity.example.id}"

  depends_on = ["aws_route53_record.example_amazonses_verification_record"]
}

```

## » Argument Reference

The following arguments are supported:

- **domain** - (Required) The domain name of the SES domain identity to verify.

## » Attributes Reference

The following attributes are exported:

- **id** - The domain name of the domain identity.
- **arn** - The ARN of the domain identity.

## » Timeouts

`acm_ses_domain_identity_verification` provides the following Timeouts configuration options:

- **create** - (Default 45m) How long to wait for a domain identity to be verified.

## » `aws__ses__domain__dkim`

Provides an SES domain DKIM generation resource.

Domain ownership needs to be confirmed first using `ses__domain__identity` Resource



## » Argument Reference

The following arguments are supported:

- **domain** - (Required) Verified domain name to generate DKIM tokens for.

## » Attributes Reference

The following attributes are exported:

- **dkim\_tokens** - DKIM tokens generated by SES. These tokens should be used to create CNAME records used to verify SES Easy DKIM. See below for an example of how this might be achieved when the domain is hosted in Route 53 and managed by Terraform. Find out more about verifying domains in Amazon SES in the AWS SES docs.

## » Example Usage

```
resource "aws_ses_domain_identity" "example" {
  domain = "example.com"
}

resource "aws_ses_domain_dkim" "example" {
  domain = "${aws_ses_domain_identity.example.domain}"
}

resource "aws_route53_record" "example_amazonses_verification_record" {
  count    = 3
  zone_id = "ABCDEFGHJIJ123"
  name     = "${element(aws_ses_domain_dkim.example.dkim_tokens, count.index)}._domainkey.example.com"
  type     = "CNAME"
  ttl      = "600"
  records  = ["${element(aws_ses_domain_dkim.example.dkim_tokens, count.index)}.dkim.amazonses.example.com"]
}
```

## » Import

DKIM tokens can be imported using the `domain` attribute, e.g.

```
$ terraform import aws_ses_domain_dkim.example example.com
```

## » `aws_ses_domain_mail_from`

Provides an SES domain MAIL FROM resource.

**NOTE:** For the MAIL FROM domain to be fully usable, this resource should be paired with the `aws_ses_domain_identity` resource. To validate the MAIL FROM domain, a DNS MX record is required. To pass SPF checks, a DNS TXT record may also be required. See the Amazon SES MAIL FROM documentation for more information.

### » Example Usage

```
resource "aws_ses_domain_mail_from" "example" {
  domain          = "${aws_ses_domain_identity.example.domain}"
  mail_from_domain = "bounce.${aws_ses_domain_identity.example.domain}"
}
```

# Example SES Domain Identity

```
resource "aws_ses_domain_identity" "example" {
  domain = "example.com"
}
```

# Example Route53 MX record

```
resource "aws_route53_record" "example_ses_domain_mail_from_mx" {
  zone_id = "${aws_route53_zone.example.id}"
  name     = "${aws_ses_domain_mail_from.example.mail_from_domain}"
  type     = "MX"
  ttl      = "600"
  records  = ["10 feedback-smtp.us-east-1.amazonses.com"] # Change to the region in which `aw
}
```

# Example Route53 TXT record for SPF

```
resource "aws_route53_record" "example_ses_domain_mail_from_txt" {
  zone_id = "${aws_route53_zone.example.id}"
  name     = "${aws_ses_domain_mail_from.example.mail_from_domain}"
  type     = "TXT"
  ttl      = "600"
  records  = ["v=spf1 include:amazonses.com -all"]
}
```

### » Argument Reference

The following arguments are required:

- `domain` - (Required) Verified domain name to generate DKIM tokens for.

- **mail\_from\_domain** - (Required) Subdomain (of above domain) which is to be used as MAIL FROM address (Required for DMARC validation)

The following arguments are optional:

- **behavior\_on\_mx\_failure** - (Optional) The action that you want Amazon SES to take if it cannot successfully read the required MX record when you send an email. Defaults to `UseDefaultValue`. See the SES API documentation for more information.

## » Attributes Reference

In addition to the arguments, which are exported, the following attributes are exported:

- **id** - The domain name.

## » Import

MAIL FROM domain can be imported using the `domain` attribute, e.g.

```
$ terraform import aws_ses_domain_mail_from.example example.com
```

## » aws\_ses\_receipt\_filter

Provides an SES receipt filter resource

## » Example Usage

```
resource "aws_ses_receipt_filter" "filter" {
  name   = "block-spammer"
  cidr   = "10.10.10.10"
  policy = "Block"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the filter
- **cidr** - (Required) The IP address or address range to filter, in CIDR notation
- **policy** - (Required) Block or Allow

## » **aws\_\_ses\_\_receipt\_\_rule**

Provides an SES receipt rule resource

### » **Example Usage**

```
# Add a header to the email and store it in S3
resource "aws_ses_receipt_rule" "store" {
  name           = "store"
  rule_set_name = "default-rule-set"
  recipients     = ["karen@example.com"]
  enabled        = true
  scan_enabled   = true

  add_header_action {
    header_name = "Custom-Header"
    header_value = "Added by SES"
  }

  s3_action {
    bucket_name = "emails"
  }
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) The name of the rule
- **rule\_set\_name** - (Required) The name of the rule set
- **after** - (Optional) The name of the rule to place this rule after
- **enabled** - (Optional) If true, the rule will be enabled
- **recipients** - (Optional) A list of email addresses
- **scan\_enabled** - (Optional) If true, incoming emails will be scanned for spam and viruses
- **tls\_policy** - (Optional) Require or Optional
- **add\_header\_action** - (Optional) A list of Add Header Action blocks. Documented below.
- **bounce\_action** - (Optional) A list of Bounce Action blocks. Documented below.
- **lambda\_action** - (Optional) A list of Lambda Action blocks. Documented below.
- **s3\_action** - (Optional) A list of S3 Action blocks. Documented below.
- **sns\_action** - (Optional) A list of SNS Action blocks. Documented below.

- **stop\_action** - (Optional) A list of Stop Action blocks. Documented below.
- **workmail\_action** - (Optional) A list of WorkMail Action blocks. Documented below.

Add header actions support the following:

- **header\_name** - (Required) The name of the header to add
- **header\_value** - (Required) The value of the header to add
- **position** - (Required) The position of the action in the receipt rule

Bounce actions support the following:

- **message** - (Required) The message to send
- **sender** - (Required) The email address of the sender
- **smtp\_reply\_code** - (Required) The RFC 5321 SMTP reply code
- **status\_code** - (Optional) The RFC 3463 SMTP enhanced status code
- **topic\_arn** - (Optional) The ARN of an SNS topic to notify
- **position** - (Required) The position of the action in the receipt rule

Lambda actions support the following:

- **function\_arn** - (Required) The ARN of the Lambda function to invoke
- **invocation\_type** - (Optional) Event or RequestResponse
- **topic\_arn** - (Optional) The ARN of an SNS topic to notify
- **position** - (Required) The position of the action in the receipt rule

S3 actions support the following:

- **bucket\_name** - (Required) The name of the S3 bucket
- **kms\_key\_arn** - (Optional) The ARN of the KMS key
- **object\_key\_prefix** - (Optional) The key prefix of the S3 bucket
- **topic\_arn** - (Optional) The ARN of an SNS topic to notify
- **position** - (Required) The position of the action in the receipt rule

SNS actions support the following:

- **topic\_arn** - (Required) The ARN of an SNS topic to notify
- **position** - (Required) The position of the action in the receipt rule

Stop actions support the following:

- **scope** - (Required) The scope to apply
- **topic\_arn** - (Optional) The ARN of an SNS topic to notify
- **position** - (Required) The position of the action in the receipt rule

WorkMail actions support the following:

- **organization\_arn** - (Required) The ARN of the WorkMail organization
- **topic\_arn** - (Optional) The ARN of an SNS topic to notify
- **position** - (Required) The position of the action in the receipt rule

## » **aws\_\_ses\_\_receipt\_\_rule\_\_set**

Provides an SES receipt rule set resource

### » **Example Usage**

```
resource "aws_ses_receipt_rule_set" "main" {  
  rule_set_name = "primary-rules"  
}
```

### » **Argument Reference**

The following arguments are supported:

- `rule_set_name` - (Required) The name of the rule set

## » **aws\_\_ses\_\_configuration\_\_set**

Provides an SES configuration set resource

### » **Example Usage**

```
resource "aws_ses_configuration_set" "test" {  
  name = "some-configuration-set-test"  
}
```

### » **Argument Reference**

The following arguments are supported:

- `name` - (Required) The name of the configuration set

## » **aws\_\_ses\_\_event\_\_destination**

Provides an SES event destination

## » Example Usage

```
# Add a firehose event destination to a configuration set
resource "aws_ses_event_destination" "kinesis" {
  name                = "event-destination-kinesis"
  configuration_set_name = "${aws_ses_configuration_set.test.name}"
  enabled             = true
  matching_types      = ["bounce", "send"]

  kinesis_destination = {
    stream_arn = "${aws_kinesis_firehose_delivery_stream.test_stream.arn}"
    role_arn   = "${aws_iam_role.firehose_role.arn}"
  }
}

# CloudWatch event destination
resource "aws_ses_event_destination" "cloudwatch" {
  name                = "event-destination-cloudwatch"
  configuration_set_name = "${aws_ses_configuration_set.test.name}"
  enabled             = true
  matching_types      = ["bounce", "send"]

  cloudwatch_destination = {
    default_value = "default"
    dimension_name = "dimension"
    value_source  = "emailHeader"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the event destination
- **configuration\_set\_name** - (Required) The name of the configuration set
- **enabled** - (Optional) If true, the event destination will be enabled
- **matching\_types** - (Required) A list of matching types. May be any of "send", "reject", "bounce", "complaint", "delivery", "open", or "click".
- **cloudwatch\_destination** - (Optional) CloudWatch destination for the events
- **kinesis\_destination** - (Optional) Send the events to a kinesis firehose destination
- **sns\_destination** - (Optional) Send the events to an SNS Topic destination

**NOTE:** You can specify "cloudwatch\_destination" or "kinesis\_destination" but not both

CloudWatch Destination requires the following:

- **default\_value** - (Required) The default value for the event
- **dimension\_name** - (Required) The name for the dimension
- **value\_source** - (Required) The source for the value. It can be either "messageTag" or "emailHeader"

Kinesis Destination requires the following:

- **stream\_arn** - (Required) The ARN of the Kinesis Stream
- **role\_arn** - (Required) The ARN of the role that has permissions to access the Kinesis Stream

SNS Topic requires the following:

- **topic\_arn** - (Required) The ARN of the SNS topic

## » ses\_identity\_notification\_topic

Resource for managing SES Identity Notification Topics

### » Example Usage

```
resource "aws_ses_identity_notification_topic" "test" {
  topic_arn          = "${aws_sns_topic.example.arn}"
  notification_type = "Bounce"
  identity            = "${aws_ses_domain_identity.example.domain}"
}
```

### » Argument Reference

The following arguments are supported:

- **topic\_arn** - (Optional) The Amazon Resource Name (ARN) of the Amazon SNS topic. Can be set to "" (an empty string) to disable publishing.
- **notification\_type** - (Required) The type of notifications that will be published to the specified Amazon SNS topic. Valid Values: *Bounce*, *Complaint* or *Delivery*.
- **identity** - (Required) The identity for which the Amazon SNS topic will be set. You can specify an identity by using its name or by using its Amazon Resource Name (ARN).



## » **aws\_ses\_template**

Provides a resource to create a SES template.

### » **Example Usage**

```
resource "aws_ses_template" "MyTemplate" {
  name      = "MyTemplate"
  subject   = "Greetings, {{name}}!"
  html      = "<h1>Hello {{name}},</h1><p>Your favorite animal is {{favoriteanimal}}.</p>"
  text      = "Hello {{name}},\r\nYour favorite animal is {{favoriteanimal}}."
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) The name of the template. Cannot exceed 64 characters. You will refer to this name when you send email.
- **html** - (Optional) The HTML body of the email. Must be less than 500KB in size, including both the text and HTML parts.
- **subject** - (Optional) The subject line of the email.
- **text** - (Optional) The email body that will be visible to recipients whose email clients do not display HTML. Must be less than 500KB in size, including both the text and HTML parts.

### » **Attributes Reference**

The following additional attributes are exported:

- **id** - The name of the SES template

### » **Import**

SES templates can be imported using the template name, e.g.

```
$ terraform import aws_ses_template.MyTemplate MyTemplate
```

## » **aws\_servicecatalog\_portfolio**

Provides a resource to create a Service Catalog Portfolio.

## » Example Usage

```
resource "aws_servicecatalog_portfolio" "portfolio" {
  name = "My App Portfolio"
  description = "List of my organizations apps"
  provider_name = "Brett"
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the portfolio.
- **description** - (Required) Description of the portfolio
- **provider\_name** - (Required) Name of the person or organization who owns the portfolio.
- **tags** - (Optional) Tags to apply to the connection.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the Service Catalog Portfolio.

## » Import

Service Catalog Portfolios can be imported using the `service catalog portfolio id`, e.g.

```
$ terraform import aws_servicecatalog_portfolio.testfolio port-12344321
```

## » `aws__service__discovery__private__dns__namespace`

Provides a Service Discovery Private DNS Namespace resource.

## » Example Usage

```
resource "aws_vpc" "example" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_service_discovery_private_dns_namespace" "example" {
```

```

name = "hoge.example.local"
description = "example"
vpc = "${aws_vpc.example.id}"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the namespace.
- **vpc** - (Required) The ID of VPC that you want to associate the namespace with.
- **description** - (Optional) The description that you specify for the namespace when you create it.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of a namespace.
- **arn** - The ARN that Amazon Route 53 assigns to the namespace when you create it.
- **hosted\_zone** - The ID for the hosted zone that Amazon Route 53 creates when you create a namespace.

## » `aws_service_discovery_public_dns_namespace`

Provides a Service Discovery Public DNS Namespace resource.

## » Example Usage

```

resource "aws_service_discovery_public_dns_namespace" "example" {
  name = "hoge.example.com"
  description = "example"
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the namespace.

- **description** - (Optional) The description that you specify for the namespace when you create it.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of a namespace.
- **arn** - The ARN that Amazon Route 53 assigns to the namespace when you create it.
- **hosted\_zone** - The ID for the hosted zone that Amazon Route 53 creates when you create a namespace.

## » Import

Service Discovery Public DNS Namespace can be imported using the namespace ID, e.g.

```
$ terraform import aws_service_discovery_public_dns_namespace.example 0123456789
```

## » aws\_\_service\_\_discovery\_\_service

Provides a Service Discovery Service resource.

## » Example Usage

```
resource "aws_vpc" "example" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_service_discovery_private_dns_namespace" "example" {
  name = "example.terraform.local"
  description = "example"
  vpc = "${aws_vpc.example.id}"
}

resource "aws_service_discovery_service" "example" {
  name = "example"
  dns_config {
    namespace_id = "${aws_service_discovery_private_dns_namespace.example.id}"
    dns_records {
      ttl = 10
    }
  }
}
```

```

        type = "A"
    }
    routing_policy = "MULTIVALUE"
}

health_check_custom_config {
    failure_threshold = 1
}
}

resource "aws_service_discovery_public_dns_namespace" "example" {
    name = "example.terraform.com"
    description = "example"
}

resource "aws_service_discovery_service" "example" {
    name = "example"
    dns_config {
        namespace_id = "${aws_service_discovery_public_dns_namespace.example.id}"
        dns_records {
            ttl = 10
            type = "A"
        }
    }
    health_check_config {
        failure_threshold = 10
        resource_path = "path"
        type = "HTTP"
    }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required, ForceNew) The name of the service.
- **description** - (Optional) The description of the service.
- **dns\_config** - (Required) A complex type that contains information about the resource record sets that you want Amazon Route 53 to create when you register an instance.
- **health\_check\_config** - (Optional) A complex type that contains settings for an optional health check. Only for Public DNS namespaces.
- **health\_check\_custom\_config** - (Optional, ForceNew) A complex type that contains settings for ECS managed health checks.

#### » **dns\_config**

The following arguments are supported:

- **namespace\_id** - (Required, ForceNew) The ID of the namespace to use for DNS configuration.
- **dns\_records** - (Required) An array that contains one DnsRecord object for each resource record set.
- **routing\_policy** - (Optional) The routing policy that you want to apply to all records that Route 53 creates when you register an instance and specify the service. Valid Values: MULTIVALUE, WEIGHTED

#### » **dns\_records**

The following arguments are supported:

- **ttl** - (Required) The amount of time, in seconds, that you want DNS resolvers to cache the settings for this resource record set.
- **type** - (Required, ForceNew) The type of the resource, which indicates the value that Amazon Route 53 returns in response to DNS queries. Valid Values: A, AAAA, SRV, CNAME

#### » **health\_check\_config**

The following arguments are supported:

- **failure\_threshold** - (Optional) The number of consecutive health checks. Maximum value of 10.
- **resource\_path** - (Optional) The path that you want Route 53 to request when performing health checks. Route 53 automatically adds the DNS name for the service. If you don't specify a value, the default value is /.
- **type** - (Optional, ForceNew) The type of health check that you want to create, which indicates how Route 53 determines whether an endpoint is healthy. Valid Values: HTTP, HTTPS, TCP

#### » **health\_check\_custom\_config**

The following arguments are supported:

- **failure\_threshold** - (Optional, ForceNew) The number of 30-second intervals that you want service discovery to wait before it changes the health status of a service instance. Maximum value of 10.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the service.
- `arn` - The ARN of the service.

## » Import

Service Discovery Service can be imported using the service ID, e.g.

```
$ terraform import aws_service_discovery_service.example 0123456789
```

## » `sfn__activity`

Provides a Step Function Activity resource

## » Example Usage

```
resource "aws_sfn_activity" "sfn_activity" {  
  name = "my-activity"  
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the activity to create.

## » Attributes Reference

The following attributes are exported:

- `id` - The Amazon Resource Name (ARN) that identifies the created activity.
- `name` - The name of the activity.
- `creation_date` - The date the activity was created.

## » Import

Activities can be imported using the `arn`, e.g.

```
$ terraform import aws_sfn_activity.foo arn:aws:states:eu-west-1:123456789098:activity:bar
```

## » `sfn__state__machine`

Provides a Step Function State Machine resource

## » Example Usage

```
# ...
```

```
resource "aws_sfn_state_machine" "sfn_state_machine" {
  name      = "my-state-machine"
  role_arn = "${aws_iam_role.iam_for_sfn.arn}"

  definition = <<EOF
{
  "Comment": "A Hello World example of the Amazon States Language using an AWS Lambda Function",
  "StartAt": "HelloWorld",
  "States": {
    "HelloWorld": {
      "Type": "Task",
      "Resource": "${aws_lambda_function.lambda.arn}",
      "End": true
    }
  }
}
EOF
}
```

## » Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the state machine.
- `definition` - (Required) The Amazon States Language definition of the state machine.
- `role_arn` - (Required) The Amazon Resource Name (ARN) of the IAM role to use for this state machine.



## » Attributes Reference

The following attributes are exported:

- **id** - The ARN of the state machine.
- **creation\_date** - The date the state machine was created.
- **status** - The current status of the state machine. Either "ACTIVE" or "DELETING".

## » Import

State Machines can be imported using the **arn**, e.g.

```
$ terraform import aws_sfn_state_machine.foo arn:aws:states:eu-west-1:123456789098:stateMach
```

## » aws\_simplifiedb\_domain

Provides a SimpleDB domain resource

## » Example Usage

```
resource "aws_simplifiedb_domain" "users" {  
  name = "users"  
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the SimpleDB domain

## » Attributes Reference

The following attributes are exported:

- **id** - The name of the SimpleDB domain

## » Import

SimpleDB Domains can be imported using the **name**, e.g.

```
$ terraform import aws_simplifiedb_domain.users users
```

## » `aws_sns_platform_application`

Provides an SNS platform application resource

### » Example Usage

#### » Apple Push Notification Service (APNS)

```
resource "aws_sns_platform_application" "apns_application" {
  name           = "apns_application"
  platform       = "APNS"
  platform_credential = "<APNS PRIVATE KEY>"
  platform_principal = "<APNS CERTIFICATE>"
}
```

#### » Google Cloud Messaging (GCM)

```
resource "aws_sns_platform_application" "gcm_application" {
  name           = "gcm_application"
  platform       = "GCM"
  platform_credential = "<GCM API KEY>"
}
```

### » Argument Reference

The following arguments are supported:

- `name` - (Required) The friendly name for the SNS platform application
- `platform` - (Required) The platform that the app is registered with. See Platform for supported platforms.
- `platform_credential` - (Required) Application Platform credential. See Credential for type of credential required for platform. The value of this attribute when stored into the Terraform state is only a hash of the real value, so therefore it is not practical to use this as an attribute for other resources.
- `event_delivery_failure_topic_arn` - (Optional) SNS Topic triggered when a delivery to any of the platform endpoints associated with your platform application encounters a permanent failure.
- `event_endpoint_created_topic_arn` - (Optional) SNS Topic triggered when a new platform endpoint is added to your platform application.
- `event_endpoint_deleted_topic_arn` - (Optional) SNS Topic triggered when an existing platform endpoint is deleted from your platform application.

- **event\_endpoint\_updated\_topic** - (Optional) SNS Topic triggered when an existing platform endpoint is changed from your platform application.
- **failure\_feedback\_role\_arn** - (Optional) The IAM role permitted to receive failure feedback for this application.
- **platform\_principal** - (Optional) Application Platform principal. See Principal for type of principal required for platform. The value of this attribute when stored into the Terraform state is only a hash of the real value, so therefore it is not practical to use this as an attribute for other resources.
- **success\_feedback\_role\_arn** - (Optional) The IAM role permitted to receive success feedback for this application.
- **success\_feedback\_sample\_rate** - (Optional) The percentage of success to sample (0-100)

## » Attributes Reference

The following additional attributes are exported:

- `id` - The ARN of the SNS platform application
- `arn` - The ARN of the SNS platform application

## » Import

SNS platform applications can be imported using the ARN, e.g.

```
$ terraform import aws_sns_platform_application.gcm_application arn:aws:sns:us-west-2:012345
```

```
» aws__sns__topic
```

Provides an SNS topic resource

## » Example Usage

```
resource "aws_sns_topic" "user_updates" {
  name = "user-updates-topic"
}
```

## » Message Delivery Status Arguments

The `<endpoint>_success_feedback_role_arn` and `<endpoint>_failure_feedback_role_arn` arguments are used to give Amazon SNS write access to use CloudWatch Logs on your behalf. The `<endpoint>_success_feedback_sample_rate` argument

is for specifying the sample rate percentage (0-100) of successfully delivered messages. After you configure the `<endpoint>_failure_feedback_role_arn` argument, then all failed message deliveries generate CloudWatch Logs.

## » Argument Reference

The following arguments are supported:

- `name` - (Optional) The friendly name for the SNS topic. By default generated by Terraform.
- `name_prefix` - (Optional) The friendly name for the SNS topic. Conflicts with `name`.
- `display_name` - (Optional) The display name for the SNS topic
- `policy` - (Optional) The fully-formed AWS policy as JSON
- `delivery_policy` - (Optional) The SNS delivery policy
- `application_success_feedback_role_arn` - (Optional) The IAM role permitted to receive success feedback for this topic
- `application_success_feedback_sample_rate` - (Optional) Percentage of success to sample
- `application_failure_feedback_role_arn` - (Optional) IAM role for failure feedback
- `http_success_feedback_role_arn` - (Optional) The IAM role permitted to receive success feedback for this topic
- `http_success_feedback_sample_rate` - (Optional) Percentage of success to sample
- `http_failure_feedback_role_arn` - (Optional) IAM role for failure feedback
- `lambda_success_feedback_role_arn` - (Optional) The IAM role permitted to receive success feedback for this topic
- `lambda_success_feedback_sample_rate` - (Optional) Percentage of success to sample
- `lambda_failure_feedback_role_arn` - (Optional) IAM role for failure feedback
- `sqs_success_feedback_role_arn` - (Optional) The IAM role permitted to receive success feedback for this topic
- `sqs_success_feedback_sample_rate` - (Optional) Percentage of success to sample
- `sqs_failure_feedback_role_arn` - (Optional) IAM role for failure feedback

## » Attributes Reference

The following attributes are exported:

- `id` - The ARN of the SNS topic

- `arn` - The ARN of the SNS topic, as a more obvious property (clone of `id`)

## » Import

SNS Topics can be imported using the `topic_arn`, e.g.

```
$ terraform import aws_sns_topic.user_updates arn:aws:sns:us-west-2:0123456789012:my-topic
```

## » `aws_sns_topic_policy`

Provides an SNS topic policy resource

**NOTE:** If a Principal is specified as just an AWS account ID rather than an ARN, AWS silently converts it to the ARN for the root user, causing future terraform plans to differ. To avoid this problem, just specify the full ARN, e.g. `arn:aws:iam::123456789012:root`

## » Example Usage

```
resource "aws_sns_topic" "test" {
  name = "my-topic-with-policy"
}

resource "aws_sns_topic_policy" "default" {
  arn = "${aws_sns_topic.test.arn}"

  policy = "${data.aws_iam_policy_document.sns-topic-policy.json}"
}

data "aws_iam_policy_document" "sns-topic-policy" {
  policy_id = "__default_policy_ID"

  statement {
    actions = [
      "SNS:Subscribe",
      "SNS:SetTopicAttributes",
      "SNS:RemovePermission",
      "SNS:Receive",
      "SNS:Publish",
      "SNS:ListSubscriptionsByTopic",
      "SNS:GetTopicAttributes",
      "SNS:DeleteTopic",
      "SNS:AddPermission",
    ]
  }
}
```

```

]

condition {
  test      = "StringEquals"
  variable = "AWS:SourceOwner"

  values = [
    "${var.account-id}",
  ]
}

effect = "Allow"

principals {
  type          = "AWS"
  identifiers = ["*"]
}

resources = [
  "${aws_sns_topic.test.arn}",
]

sid = "__default_statement_ID"
}

```

## » Argument Reference

The following arguments are supported:

- **arn** - (Required) The ARN of the SNS topic
- **policy** - (Required) The fully-formed AWS policy as JSON

## » `aws__sns__topic__subscription`

Provides a resource for subscribing to SNS topics. Requires that an SNS topic exist for the subscription to attach to. This resource allows you to automatically place messages sent to SNS topics in SQS queues, send them as HTTP(S) POST requests to a given endpoint, send SMS messages, or notify devices / applications. The most likely use case for Terraform users will probably be SQS queues.

**NOTE:** If SNS topic and SQS queue are in different AWS regions it is important to place the "aws\_\_sns\_\_topic\_\_subscription" into the terraform configuration of the region with the SQS queue. If "aws\_\_sns\_\_topic\_\_subscription" is placed in

the terraform configuration of the region with the SNS topic terraform will fail to create the subscription.

**NOTE:** Setup of cross-account subscriptions from SNS topics to SQS queues requires Terraform to have access to BOTH accounts.

**NOTE:** If SNS topic and SQS queue are in different AWS accounts but the same region it is important to place the "aws\_sns\_topic\_subscription" into the terraform configuration of the account with the SQS queue. If "aws\_sns\_topic\_subscription" is placed in the terraform configuration of the account with the SNS topic terraform creates the subscriptions but does not keep state and tries to re-create the subscription at every apply.

**NOTE:** If SNS topic and SQS queue are in different AWS accounts and different AWS regions it is important to recognize that the subscription needs to be initiated from the account with the SQS queue but in the region of the SNS topic.

## » Example Usage

You can directly supply a topic and ARN by hand in the `topic_arn` property along with the queue ARN:

```
resource "aws_sns_topic_subscription" "user_updates_sqs_target" {
  topic_arn = "arn:aws:sns:us-west-2:432981146916:user-updates-topic"
  protocol  = "sqs"
  endpoint  = "arn:aws:sqs:us-west-2:432981146916:terraform-queue-too"
}
```

Alternatively you can use the ARN properties of a managed SNS topic and SQS queue:

```
resource "aws_sns_topic" "user_updates" {
  name = "user-updates-topic"
}

resource "aws_sqs_queue" "user_updates_queue" {
  name = "user-updates-queue"
}

resource "aws_sns_topic_subscription" "user_updates_sqs_target" {
  topic_arn = "${aws_sns_topic.user_updates.arn}"
  protocol  = "sqs"
  endpoint  = "${aws_sqs_queue.user_updates_queue.arn}"
}
```

You can subscribe SNS topics to SQS queues in different Amazon accounts and regions:

```

/*
#
# Variables
#
*/
variable "sns" {
  default = {
    account-id   = "111111111111"
    role-name    = "service/service-hashicorp-terraform"
    name         = "example-sns-topic"
    display_name = "example"
    region       = "us-west-1"
  }
}

variable "sqs" {
  default = {
    account-id   = "222222222222"
    role-name    = "service/service-hashicorp-terraform"
    name         = "example-sqs-queue"
    region       = "us-east-1"
  }
}

data "aws_iam_policy_document" "sns-topic-policy" {
  policy_id = "__default_policy_ID"

  statement {
    actions = [
      "SNS:Subscribe",
      "SNS:SetTopicAttributes",
      "SNS:RemovePermission",
      "SNS:Receive",
      "SNS:Publish",
      "SNS:ListSubscriptionsByTopic",
      "SNS:GetTopicAttributes",
      "SNS:DeleteTopic",
      "SNS:AddPermission",
    ]

    condition {
      test      = "StringEquals"
      variable  = "AWS:SourceOwner"

      values = [
        "${var.sns["account-id"]}",

```



```

    ]
}

effect = "Allow"

principals {
    type      = "AWS"
    identifiers = ["*"]
}

resources = [
    "arn:aws:sns:${var.sns["region"]}:${var.sns["account-id"]}:${var.sns["name"]}",
]

sid = "__default_statement_ID"
}

statement {
    actions = [
        "SNS:Subscribe",
        "SNS:Receive",
    ]

    condition {
        test      = "StringLike"
        variable = "SNS:Endpoint"

        values = [
            "arn:aws:sqs:${var.sqs["region"]}:${var.sqs["account-id"]}:${var.sqs["name"]}",
        ]
    }

    effect = "Allow"

    principals {
        type      = "AWS"
        identifiers = ["*"]
    }

    resources = [
        "arn:aws:sns:${var.sns["region"]}:${var.sns["account-id"]}:${var.sns["name"]}",
    ]

    sid = "__console_sub_0"
}
}

```

```

data "aws_iam_policy_document" "sqs-queue-policy" {
  policy_id = "arn:aws:sqs:${var.sqs["region"]}:${var.sqs["account-id"]}:${var.sqs["name"]}"

  statement {
    sid      = "example-sns-topic"
    effect   = "Allow"

    principals {
      type       = "AWS"
      identifiers = ["*"]
    }

    actions = [
      "SQS:SendMessage",
    ]

    resources = [
      "arn:aws:sqs:${var.sqs["region"]}:${var.sqs["account-id"]}:${var.sqs["name"]}",
    ]

    condition {
      test      = "ArnEquals"
      variable   = "aws:SourceArn"

      values = [
        "arn:aws:sns:${var.sns["region"]}:${var.sns["account-id"]}:${var.sns["name"]}",
      ]
    }
  }
}

# provider to manage SNS topics
provider "aws" {
  alias  = "sns"
  region = "${var.sns["region"]}"

  assume_role {
    role_arn      = "arn:aws:iam::${var.sns["account-id"]}:role/${var.sns["role-name"]}"
    session_name = "sns-${var.sns["region"]}"
  }
}

# provider to manage SQS queues
provider "aws" {
  alias = "sqs"

```

```

region = "${var.sqs["region"]}"

assume_role {
  role_arn      = "arn:aws:iam:${var.sqs["account-id"]}:role/${var.sqs["role-name"]}"
  session_name = "sqs-${var.sqs["region"]}"
}

# provider to subscribe SQS to SNS (using the SQS account but the SNS region)
provider "aws" {
  alias = "sns2sqs"
  region = "${var.sns["region"]}"

  assume_role {
    role_arn      = "arn:aws:iam:${var.sqs["account-id"]}:role/${var.sqs["role-name"]}"
    session_name = "sns2sqs-${var.sns["region"]}"
  }
}

resource "aws_sns_topic" "sns-topic" {
  provider      = "aws.sns"
  name          = "${var.sns["name"]}"
  display_name = "${var.sns["display_name"]}"
  policy        = "${data.aws_iam_policy_document.sns-topic-policy.json}"
}

resource "aws_sqs_queue" "sqs-queue" {
  provider = "aws.sqs"
  name     = "${var.sqs["name"]}"
  policy   = "${data.aws_iam_policy_document.sqs-queue-policy.json}"
}

resource "aws_sns_topic_subscription" "sns-topic" {
  provider      = "aws.sns2sqs"
  topic_arn     = "${aws_sns_topic.sns-topic.arn}"
  protocol      = "sqs"
  endpoint      = "${aws_sqs_queue.sqs-queue.arn}"
}

```

## » Argument Reference

The following arguments are supported:

- `topic_arn` - (Required) The ARN of the SNS topic to subscribe to
- `protocol` - (Required) The protocol to use. The possible values for this

are: `sqs`, `sms`, `lambda`, `application`. (`http` or `https` are partially supported, see below) (`email` is option but unsupported, see below).

- `endpoint` - (Required) The endpoint to send data to, the contents will vary with the protocol. (see below for more information)
- `endpoint_auto_confirms` - (Optional) Boolean indicating whether the end point is capable of auto confirming subscription e.g., PagerDuty (default is false)
- `confirmation_timeout_in_minutes` - (Optional) Integer indicating number of minutes to wait in retrying mode for fetching subscription arn before marking it as failure. Only applicable for `http` and `https` protocols (default is 1 minute).
- `raw_message_delivery` - (Optional) Boolean indicating whether or not to enable raw message delivery (the original message is directly passed, not wrapped in JSON with the original message in the `message` property) (default is false).
- `filter_policy` - (Optional) The text of a filter policy to the topic subscription.

## » Protocols supported

Supported SNS protocols include:

- `lambda` -- delivery of JSON-encoded message to a lambda function
- `sqs` -- delivery of JSON-encoded message to an Amazon SQS queue
- `application` -- delivery of JSON-encoded message to an EndpointArn for a mobile app and device
- `sms` -- delivery text message

Partially supported SNS protocols include:

- `http` -- delivery of JSON-encoded messages via HTTP. Supported only for the end points that auto confirms the subscription.
- `https` -- delivery of JSON-encoded messages via HTTPS. Supported only for the end points that auto confirms the subscription.

Unsupported protocols include the following:

- `email` -- delivery of message via SMTP
- `email-json` -- delivery of JSON-encoded message via SMTP

These are unsupported because the endpoint needs to be authorized and does not generate an ARN until the target email address has been validated. This breaks the Terraform model and as a result are not currently supported.

## » Specifying endpoints

Endpoints have different format requirements according to the protocol that is chosen.

- SQS endpoints come in the form of the SQS queue's ARN (not the URL of the queue) e.g: `arn:aws:sqs:us-west-2:432981146916:terraform-queue-too`
- Application endpoints are also the endpoint ARN for the mobile app and device.

## » Attributes Reference

The following attributes are exported:

- `id` - The ARN of the subscription
- `topic_arn` - The ARN of the topic the subscription belongs to
- `protocol` - The protocol being used
- `endpoint` - The full endpoint to send data to (SQS ARN, HTTP(S) URL, Application ARN, SMS number, etc.)
- `arn` - The ARN of the subscription stored as a more user-friendly property

## » Import

SNS Topic Subscriptions can be imported using the `subscription_arn`, e.g.

```
$ terraform import aws_sns_topic_subscription.user_updates_sqs_target arn:aws:sns:us-west-2:
```

## » `aws__ssm__activation`

Registers an on-premises server or virtual machine with Amazon EC2 so that it can be managed using Run Command.

## » Example Usage

```
resource "aws_iam_role" "test_role" {
  name = "test_role"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Principal": {"Service": "ssm.amazonaws.com"},
```

```

        "Action": "sts:AssumeRole"
    }
}
EOF
}

resource "aws_iam_role_policy_attachment" "test_attach" {
    role      = "${aws_iam_role.test_role.name}"
    policy_arn = "arn:aws:iam::aws:policy/service-role/AmazonEC2RoleforSSM"
}

resource "aws_ssm_activation" "foo" {
    name              = "test_ssm_activation"
    description       = "Test"
    iam_role          = "${aws_iam_role.test_role.id}"
    registration_limit = "5"
    depends_on        = ["aws_iam_role_policy_attachment.test_attach"]
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) The default name of the registered managed instance.
- **description** - (Optional) The description of the resource that you want to register.
- **expiration\_date** - (Optional) A timestamp in RFC3339 format by which this activation request should expire. The default value is 24 hours from resource creation time.
- **iam\_role** - (Required) The IAM Role to attach to the managed instance.
- **registration\_limit** - (Optional) The maximum number of managed instances you want to register. The default value is 1 instance.

## » Attributes Reference

The following attributes are exported:

- **activation\_code** - The code the system generates when it processes the activation.
- **name** - The default name of the registered managed instance.
- **description** - The description of the resource that was registered.
- **expired** - If the current activation has expired.
- **expiration\_date** - The date by which this activation request should expire. The default value is 24 hours.
- **iam\_role** - The IAM Role attached to the managed instance.

- **registration\_limit** - The maximum number of managed instances you want to be registered. The default value is 1 instance.
- **registration\_count** - The number of managed instances that are currently registered using this activation.

## » **aws\_\_ssm\_\_association**

Associates an SSM Document to an instance or EC2 tag.

### » **Example Usage**

```
resource "aws_ssm_association" "example" {
  name = "${aws_ssm_document.example.name}"

  targets {
    key    = "InstanceIds"
    values = "${aws_instance.example.id}"
  }
}
```

### » **Argument Reference**

The following arguments are supported:

- **name** - (Required) The name of the SSM document to apply.
- **association\_name** - (Optional) The descriptive name for the association.
- **document\_version** - (Optional) The document version you want to associate with the target(s). Can be a specific version or the default version.
- **instance\_id** - (Optional) The instance ID to apply an SSM document to. Use **targets** with key **InstanceIds** for document schema versions 2.0 and above.
- **output\_location** - (Optional) An output location block. Output Location is documented below.
- **parameters** - (Optional) A block of arbitrary string parameters to pass to the SSM document.
- **schedule\_expression** - (Optional) A cron expression when the association will be applied to the target(s).
- **targets** - (Optional) A block containing the targets of the SSM association. Targets are documented below. AWS currently supports a maximum of 5 targets.

Output Location (**output\_location**) is an S3 bucket where you want to store the results of this association:

- `s3_bucket_name` - (Required) The S3 bucket name.
- `s3_key_prefix` - (Optional) The S3 bucket prefix. Results stored in the root if not configured.

Targets specify what instance IDs or tags to apply the document to and has these keys:

- `key` - (Required) Either `InstanceIds` or `tag:Tag Name` to specify an EC2 tag.
- `values` - (Required) A list of instance IDs or tag values. AWS currently limits this to 1 target value.

## » Attributes Reference

The following attributes are exported:

- `name` - The name of the SSM document to apply.
- `instance_ids` - The instance id that the SSM document was applied to.
- `parameters` - Additional parameters passed to the SSM document.

## » `aws_ssm_document`

Provides an SSM Document resource

**NOTE on updating SSM documents:** Only documents with a schema version of 2.0 or greater can update their content once created, see SSM Schema Features. To update a document with an older schema version you must recreate the resource.

## » Example Usage

```
resource "aws_ssm_document" "foo" {
  name           = "test_document"
  document_type  = "Command"

  content = <<DOC
{
  "schemaVersion": "1.2",
  "description": "Check ip configuration of a Linux instance.",
  "parameters": {

  },
  "runtimeConfig": {
    "aws:runShellScript": {
```



```

        "properties": [
            {
                "id": "0.aws:runShellScript",
                "runCommand": ["ifconfig"]
            }
        ]
    }
}
DOC
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the document.
- **content** - (Required) The JSON or YAML content of the document.
- **document\_format** - (Optional, defaults to JSON) The format of the document. Valid document types include: **JSON** and **YAML**
- **document\_type** - (Required) The type of the document. Valid document types include: **Command**, **Policy** and **Automation**
- **permissions** - (Optional) Additional Permissions to attach to the document. See Permissions below for details.

## » Attributes Reference

The following additional attributes are exported:

- **created\_date** - The date the document was created.
- **description** - The description of the document.
- **schema\_version** - The schema version of the document.
- **default\_version** - The default version of the document.
- **hash** - The sha1 or sha256 of the document content
- **hash\_type** - "Sha1" "Sha256". The hashing algorithm used when hashing the content.
- **latest\_version** - The latest version of the document.
- **owner** - The AWS user account of the person who created the document.
- **status** - "Creating", "Active" or "Deleting". The current status of the document.
- **parameter** - The parameters that are available to this document.
- **platform\_types** - A list of OS platforms compatible with this SSM document, either "Windows" or "Linux".

## » Permissions

The permissions attribute specifies how you want to share the document. If you share a document privately, you must specify the AWS user account IDs for those people who can use the document. If you share a document publicly, you must specify All as the account ID.

The permissions mapping supports the following:

- **type** - The permission type for the document. The permission type can be **Share**.
- **account\_ids** - The AWS user accounts that should have access to the document. The account IDs can either be a group of account IDs or **All**.

## » `aws_ssm_maintenance_window`

Provides an SSM Maintenance Window resource

## » Example Usage

```
resource "aws_ssm_maintenance_window" "production" {
  name = "maintenance-window-application"
  schedule = "cron(0 16 ? * TUE *)"
  duration = 3
  cutoff = 1
}
```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the maintenance window.
- **schedule** - (Required) The schedule of the Maintenance Window in the form of a cron or rate expression.
- **cutoff** - (Required) The number of hours before the end of the Maintenance Window that Systems Manager stops scheduling new tasks for execution.
- **duration** - (Required) The duration of the Maintenance Window in hours.
- **allow\_unassociated\_targets** - (Optional) Whether targets must be registered with the Maintenance Window before tasks can be defined for those targets.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the maintenance window.

## » `aws_ssm_maintenance_window_target`

Provides an SSM Maintenance Window Target resource

## » Example Usage

```
resource "aws_ssm_maintenance_window" "window" {
  name = "maintenance-window-webapp"
  schedule = "cron(0 16 ? * TUE *)"
  duration = 3
  cutoff = 1
}

resource "aws_ssm_maintenance_window_target" "target1" {
  window_id = "${aws_ssm_maintenance_window.window.id}"
  resource_type = "INSTANCE"
  targets {
    key = "tag:Name"
    values = ["acceptance_test"]
  }
}
```

## » Argument Reference

The following arguments are supported:

- `window_id` - (Required) The Id of the maintenance window to register the target with.
- `resource_type` - (Required) The type of target being registered with the Maintenance Window. Possible values `INSTANCE`.
- `targets` - (Required) The targets (either instances or tags). Instances are specified using `Key=instanceids,Values=instanceid1,instanceid2`. Tags are specified using `Key=tag name,Values=tag value`.
- `owner_information` - (Optional) User-provided value that will be included in any CloudWatch events raised while running tasks for these targets in this Maintenance Window.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the maintenance window target.

## » `aws_ssm_maintenance_window_task`

Provides an SSM Maintenance Window Task resource

## » Example Usage

```
resource "aws_ssm_maintenance_window" "window" {
  name = "maintenance-window-%s"
  schedule = "cron(0 16 ? * TUE *)"
  duration = 3
  cutoff = 1
}

resource "aws_ssm_maintenance_window_task" "task" {
  window_id = "${aws_ssm_maintenance_window.window.id}"
  task_type = "RUN_COMMAND"
  task_arn = "AWS-RunShellScript"
  priority = 1
  service_role_arn = "arn:aws:iam::187416307283:role/service-role/AWS_Events_Invoke_Run_Command"
  max_concurrency = "2"
  max_errors = "1"
  targets {
    key = "InstanceIds"
    values = ["${aws_instance.instance.id}"]
  }
  task_parameters {
    name = "commands"
    values = ["pwd"]
  }
}

resource "aws_instance" "instance" {
  ami = "ami-4fccb37f"

  instance_type = "m1.small"
}
```

## » Argument Reference

The following arguments are supported:

- **window\_id** - (Required) The Id of the maintenance window to register the task with.
- **max\_concurrency** - (Required) The maximum number of targets this task can be run for in parallel.
- **max\_errors** - (Required) The maximum number of errors allowed before this task stops being scheduled.
- **task\_type** - (Required) The type of task being registered. The only allowed value is `RUN_COMMAND`.
- **task\_arn** - (Required) The ARN of the task to execute.
- **service\_role\_arn** - (Required) The role that should be assumed when executing the task.
- **targets** - (Required) The targets (either instances or window target ids). Instances are specified using `Key=InstanceIds,Values=instanceid1,instanceid2`. Window target ids are specified using `Key=WindowTargetIds,Values=window target id1, window target id2`.
- **priority** - (Optional) The priority of the task in the Maintenance Window, the lower the number the higher the priority. Tasks in a Maintenance Window are scheduled in priority order with tasks that have the same priority scheduled in parallel.
- **logging\_info** - (Optional) A structure containing information about an Amazon S3 bucket to write instance-level logs to. Documented below.
- **task\_parameters** - (Optional) A structure containing information about parameters required by the particular **task\_arn**. Documented below.

**logging\_info** supports the following:

- **s3\_bucket\_name** - (Required)
- **s3\_region** - (Required)
- **s3\_bucket\_prefix** - (Optional)

**task\_parameters** supports the following:

- **name** - (Required)
- **values** - (Required)

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the maintenance window task.

## » aws\_ssm\_patch\_baseline

Provides an SSM Patch Baseline resource

**NOTE on Patch Baselines:** The `approved_patches` and `approval_rule` are both marked as optional fields, but the Patch Baseline requires that at least one of them is specified.

### » Example Usage

Basic usage using `approved_patches` only

```
resource "aws_ssm_patch_baseline" "production" {
  name     = "patch-baseline"
  approved_patches = ["KB123456"]
}
```

Advanced usage, specifying patch filters

```
resource "aws_ssm_patch_baseline" "production" {
  name           = "patch-baseline"
  description    = "Patch Baseline Description"
  approved_patches = ["KB123456", "KB456789"]
  rejected_patches = ["KB987654"]
  global_filter {
    key = "PRODUCT"
    values = ["WindowsServer2008"]
  }
  global_filter {
    key = "CLASSIFICATION"
    values = ["ServicePacks"]
  }
  global_filter {
    key = "MSRC_SEVERITY"
    values = ["Low"]
  }
  approval_rule {
    approve_after_days = 7
    compliance_level = "HIGH"
    patch_filter {
      key = "PRODUCT"
      values = ["WindowsServer2016"]
    }
    patch_filter {
      key = "CLASSIFICATION"
      values = ["CriticalUpdates", "SecurityUpdates", "Updates"]
    }
  }
}
```

```

    }
    patch_filter {
        key = "MSRC_SEVERITY"
        values = ["Critical", "Important", "Moderate"]
    }
}
approval_rule {
    approve_after_days = 7
    patch_filter {
        key = "PRODUCT"
        values = ["WindowsServer2012"]
    }
}
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the patch baseline.
- **description** - (Optional) The description of the patch baseline.
- **operating\_system** - (Optional) Defines the operating system the patch baseline applies to. Supported operating systems include `WINDOWS`, `AMAZON_LINUX`, `UBUNTU` and `REDHAT_ENTERPRISE_LINUX`. The Default value is `WINDOWS`.
- **approved\_patches\_compliance\_level** - (Optional) Defines the compliance level for approved patches. This means that if an approved patch is reported as missing, this is the severity of the compliance violation. Valid compliance levels include the following: `CRITICAL`, `HIGH`, `MEDIUM`, `LOW`, `INFORMATIONAL`, `UNSPECIFIED`. The default value is `UNSPECIFIED`.
- **approved\_patches** - (Optional) A list of explicitly approved patches for the baseline.
- **rejected\_patches** - (Optional) A list of rejected patches.
- **global\_filter** - (Optional) A set of global filters used to exclude patches from the baseline. Up to 4 global filters can be specified using Key/Value pairs. Valid Keys are `PRODUCT` | `CLASSIFICATION` | `MSRC_SEVERITY` | `PATCH_ID`.
- **approval\_rule** - (Optional) A set of rules used to include patches in the baseline. up to 10 approval rules can be specified. Each `approval_rule` block requires the fields documented below.

The `approval_rule` block supports:

- **approve\_after\_days** - (Required) The number of days after the release date of each patch matched by the rule the patch is marked as approved in the patch baseline. Valid Range: 0 to 100.

- **patch\_filter** - (Required) The patch filter group that defines the criteria for the rule. Up to 4 patch filters can be specified per approval rule using Key/Value pairs. Valid Keys are **PRODUCT** | **CLASSIFICATION** | **MSRC\_SEVERITY** | **PATCH\_ID**.
- **compliance\_level** - (Optional) Defines the compliance level for patches approved by this rule. Valid compliance levels include the following: **CRITICAL**, **HIGH**, **MEDIUM**, **LOW**, **INFORMATIONAL**, **UNSPECIFIED**. The default value is **UNSPECIFIED**.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the patch baseline.

## » **aws\_ssm\_patch\_group**

Provides an SSM Patch Group resource

## » Example Usage

```
resource "aws_ssm_patch_baseline" "production" {
  name      = "patch-baseline"
  approved_patches = ["KB123456"]
}

resource "aws_ssm_patch_group" "patchgroup" {
  baseline_id = "${aws_ssm_patch_baseline.production.id}"
  patch_group = "patch-group-name"
}
```

## » Argument Reference

The following arguments are supported:

- **baseline\_id** - (Required) The ID of the patch baseline to register the patch group with.
- **patch\_group** - (Required) The name of the patch group that should be registered with the patch baseline.



## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the patch baseline.

## » `aws_ssm_parameter`

Provides an SSM Parameter resource.

## » Example Usage

To store a basic string parameter:

```
resource "aws_ssm_parameter" "foo" {
  name   = "foo"
  type   = "String"
  value  = "bar"
}
```

To store an encrypted string using the default SSM KMS key:

```
resource "aws_db_instance" "default" {
  allocated_storage    = 10
  storage_type         = "gp2"
  engine               = "mysql"
  engine_version       = "5.7.16"
  instance_class       = "db.t2.micro"
  name                 = "mydb"
  username             = "foo"
  password             = "${var.database_master_password}"
  db_subnet_group_name = "my_database_subnet_group"
  parameter_group_name = "default.mysql5.7"
}
```

```
resource "aws_ssm_parameter" "secret" {
  name       = "${var.environment}/database/password/master"
  description = "The parameter description"
  type       = "SecureString"
  value      = "${var.database_master_password}"

  tags {
    environment = "${var.environment}"
  }
}
```

**Note:** The unencrypted value of a `SecureString` will be stored in the raw state as plain-text. Read more about sensitive data in state.

## » Argument Reference

The following arguments are supported:

- **name** - (Required) The name of the parameter.
- **type** - (Required) The type of the parameter. Valid types are `String`, `StringList` and `SecureString`.
- **value** - (Required) The value of the parameter.
- **description** - (Optional) The description of the parameter.
- **key\_id** - (Optional) The KMS key id or arn for encrypting a `SecureString`.
- **overwrite** - (Optional) Overwrite an existing parameter. If not specified, will default to `false` if the resource has not been created by terraform to avoid overwrite of existing resource and will default to `true` otherwise (terraform lifecycle rules should then be used to manage the update behavior).
- **allowed\_pattern** - (Optional) A regular expression used to validate the parameter value.
- **tags** - (Optional) A mapping of tags to assign to the object.

## » Attributes Reference

The following attributes are exported:

- **arn** - The ARN of the parameter.
- **name** - (Required) The name of the parameter.
- **description** - (Required) The description of the parameter.
- **type** - (Required) The type of the parameter. Valid types are `String`, `StringList` and `SecureString`.
- **value** - (Required) The value of the parameter.

## » Import

SSM Parameters can be imported using the `parameter store name`, e.g.

```
$ terraform import aws_ssm_parameter.my_param /my_path/my_paramname
```

## » `aws__athena__database`

Provides a SSM resource data sync.

## » Example Usage

```
resource "aws_s3_bucket" "hoge" {
  bucket = "tf-test-bucket-1234"
  region = "us-east-1"
}

resource "aws_s3_bucket_policy" "hoge" {
  bucket = "${aws_s3_bucket.hoge.bucket}"
  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "SSMBucketPermissionsCheck",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "s3:GetBucketAcl",
      "Resource": "arn:aws:s3:::tf-test-bucket-1234"
    },
    {
      "Sid": " SSMBucketDelivery",
      "Effect": "Allow",
      "Principal": {
        "Service": "ssm.amazonaws.com"
      },
      "Action": "s3:PutObject",
      "Resource": ["arn:aws:s3:::tf-test-bucket-1234/*"],
      "Condition": {
        "StringEquals": {
          "s3:x-amz-acl": "bucket-owner-full-control"
        }
      }
    }
  ]
}
EOF
}

resource "aws_ssm_resource_data_sync" "foo" {
  name = "foo"
  s3_destination = {
    bucket_name = "${aws_s3_bucket.hoge.bucket}"
  }
}
```

```

    region = "${aws_s3_bucket.hoge.region}"
  }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Required) Name for the configuration.
- **s3\_destination** - (Required) Amazon S3 configuration details for the sync.

### » s3\_destination

s3\_destination supports the following:

- **bucket\_name** - (Required) Name of S3 bucket where the aggregated data is stored.
- **region** - (Required) Region with the bucket targeted by the Resource Data Sync.
- **kms\_key\_arn** - (Optional) ARN of an encryption key for a destination in Amazon S3.
- **prefix** - (Optional) Prefix for the bucket.
- **sync\_format** - (Optional) A supported sync format. Only JsonSerializerDe is currently supported. Defaults to JsonSerializerDe.

## » Import

SSM resource data sync can be imported using the **name**, e.g.

```
$ terraform import aws_ssm_resource_data_sync.example example-name
```

## » aws\_sqs\_queue

### » Example Usage

```

resource "aws_sqs_queue" "terraform_queue" {
  name                  = "terraform-example-queue"
  delay_seconds         = 90
  max_message_size     = 2048
  message_retention_seconds = 86400
  receive_wait_time_seconds = 10
  redrive_policy        = "{\"deadLetterTargetArn\":\"${aws_sqs_queue.terraform_queue_d

```

```

tags {
  Environment = "production"
}
}

```

## » FIFO queue

```

resource "aws_sqs_queue" "terraform_queue" {
  name                = "terraform-example-queue.fifo"
  fifo_queue          = true
  content_based_deduplication = true
}

```

## » Server-side encryption (SSE)

```

resource "aws_sqs_queue" "terraform_queue" {
  name                = "terraform-example-queue"
  kms_master_key_id   = "alias/aws/sqs"
  kms_data_key_reuse_period_seconds = 300
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional) This is the human-readable name of the queue. If omitted, Terraform will assign a random name.
- **name\_prefix** - (Optional) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **visibility\_timeout\_seconds** - (Optional) The visibility timeout for the queue. An integer from 0 to 43200 (12 hours). The default for this attribute is 30. For more information about visibility timeout, see AWS docs.
- **message\_retention\_seconds** - (Optional) The number of seconds Amazon SQS retains a message. Integer representing seconds, from 60 (1 minute) to 1209600 (14 days). The default for this attribute is 345600 (4 days).
- **max\_message\_size** - (Optional) The limit of how many bytes a message can contain before Amazon SQS rejects it. An integer from 1024 bytes (1 KiB) up to 262144 bytes (256 KiB). The default for this attribute is 262144 (256 KiB).

- **delay\_seconds** - (Optional) The time in seconds that the delivery of all messages in the queue will be delayed. An integer from 0 to 900 (15 minutes). The default for this attribute is 0 seconds.
- **receive\_wait\_time\_seconds** - (Optional) The time for which a `ReceiveMessage` call will wait for a message to arrive (long polling) before returning. An integer from 0 to 20 (seconds). The default for this attribute is 0, meaning that the call will return immediately.
- **policy** - (Optional) The JSON policy for the SQS queue
- **redrive\_policy** - (Optional) The JSON policy to set up the Dead Letter Queue, see AWS docs. **Note:** when specifying `maxReceiveCount`, you must specify it as an integer (5), and not a string ("5").
- **fifo\_queue** - (Optional) Boolean designating a FIFO queue. If not set, it defaults to `false` making it standard.
- **content\_based\_deduplication** - (Optional) Enables content-based deduplication for FIFO queues. For more information, see the related documentation
- **kms\_master\_key\_id** - (Optional) The ID of an AWS-managed customer master key (CMK) for Amazon SQS or a custom CMK. For more information, see Key Terms.
- **kms\_data\_key\_reuse\_period\_seconds** - (Optional) The length of time, in seconds, for which Amazon SQS can reuse a data key to encrypt or decrypt messages before calling AWS KMS again. An integer representing seconds, between 60 seconds (1 minute) and 86,400 seconds (24 hours). The default is 300 (5 minutes).
- **tags** - (Optional) A mapping of tags to assign to the queue.

## » Attributes Reference

The following attributes are exported:

- **id** - The URL for the created Amazon SQS queue.
- **arn** - The ARN of the SQS queue

## » Import

SQS Queues can be imported using the `queue_url`, e.g.

```
$ terraform import aws_sqs_queue.public_queue https://queue.amazonaws.com/80398EXAMPLE/MyQueue
```

## » aws\_sqs\_queue\_policy

Allows you to set a policy of an SQS Queue while referencing ARN of the queue within the policy.

## » Example Usage

```
resource "aws_sqs_queue" "q" {
  name = "examplequeue"
}

resource "aws_sqs_queue_policy" "test" {
  queue_url = "${aws_sqs_queue.q.id}"

  policy = <<POLICY
{
  "Version": "2012-10-17",
  "Id": "sqspolicy",
  "Statement": [
    {
      "Sid": "First",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "sqs:SendMessage",
      "Resource": "${aws_sqs_queue.q.arn}",
      "Condition": {
        "ArnEquals": {
          "aws:SourceArn": "${aws_sqs_queue.q.arn}"
        }
      }
    }
  ]
}
POLICY
}
```

## » Argument Reference

The following arguments are supported:

- `queue_url` - (Required) The URL of the SQS Queue to which to attach the policy
- `policy` - (Required) The JSON policy for the SQS queue

## » Import

SQS Queue Policies can be imported using the queue URL, e.g.

```
$ terraform import aws_sqs_queue_policy.test https://queue.amazonaws.com/0123456789012/myqueue
```

## » **aws\_\_customer\_\_gateway**

Provides a customer gateway inside a VPC. These objects can be connected to VPN gateways via VPN connections, and allow you to establish tunnels between your network and the VPC.

### » **Example Usage**

```
resource "aws_customer_gateway" "main" {
  bgp_asn      = 65000
  ip_address   = "172.83.124.10"
  type         = "ipsec.1"

  tags {
    Name = "main-customer-gateway"
  }
}
```

### » **Argument Reference**

The following arguments are supported:

- **bgp\_asn** - (Required) The gateway's Border Gateway Protocol (BGP) Autonomous System Number (ASN).
- **ip\_address** - (Required) The IP address of the gateway's Internet-routable external interface.
- **type** - (Required) The type of customer gateway. The only type AWS supports at this time is "ipsec.1".
- **tags** - (Optional) Tags to apply to the gateway.

### » **Attribute Reference**

The following attributes are exported:

- **id** - The amazon-assigned ID of the gateway.
- **bgp\_asn** - The gateway's Border Gateway Protocol (BGP) Autonomous System Number (ASN).
- **ip\_address** - The IP address of the gateway's Internet-routable external interface.
- **type** - The type of customer gateway.
- **tags** - Tags applied to the gateway.



## » Import

Customer Gateways can be imported using the `id`, e.g.

```
$ terraform import aws_customer_gateway.main cgw-b4dc3961
```

## » `aws_default_network_acl`

Provides a resource to manage the default AWS Network ACL. VPC Only.

Each VPC created in AWS comes with a Default Network ACL that can be managed, but not destroyed. **This is an advanced resource**, and has special caveats to be aware of when using it. Please read this document in its entirety before using this resource.

The `aws_default_network_acl` behaves differently from normal resources, in that Terraform does not *create* this resource, but instead attempts to "adopt" it into management. We can do this because each VPC created has a Default Network ACL that cannot be destroyed, and is created with a known set of default rules.

When Terraform first adopts the Default Network ACL, it **immediately removes all rules in the ACL**. It then proceeds to create any rules specified in the configuration. This step is required so that only the rules specified in the configuration are created.

This resource treats its inline rules as absolute; only the rules defined inline are created, and any additions/removals external to this resource will result in diffs being shown. For these reasons, this resource is incompatible with the `aws_network_acl_rule` resource.

For more information about Network ACLs, see the AWS Documentation on Network ACLs.

## » Basic Example Usage, with default rules

The following config gives the Default Network ACL the same rules that AWS includes, but pulls the resource under management by Terraform. This means that any ACL rules added or changed will be detected as drift.

```
resource "aws_vpc" "mainvpc" {
  cidr_block = "10.1.0.0/16"
}

resource "aws_default_network_acl" "default" {
  default_network_acl_id = "${aws_vpc.mainvpc.default_network_acl_id}"
}
```

```

ingress {
    protocol    = -1
    rule_no     = 100
    action      = "allow"
    cidr_block  = "0.0.0.0/0"
    from_port   = 0
    to_port     = 0
}

egress {
    protocol    = -1
    rule_no     = 100
    action      = "allow"
    cidr_block  = "0.0.0.0/0"
    from_port   = 0
    to_port     = 0
}
}

```

## » Example config to deny all Egress traffic, allowing Ingress

The following denies all Egress traffic by omitting any `egress` rules, while including the default `ingress` rule to allow all traffic.

```

resource "aws_vpc" "mainvpc" {
    cidr_block = "10.1.0.0/16"
}

resource "aws_default_network_acl" "default" {
    default_network_acl_id = "${aws_vpc.mainvpc.default_network_acl_id}"

    ingress {
        protocol    = -1
        rule_no     = 100
        action      = "allow"
        cidr_block  = "0.0.0.0/0"
        from_port   = 0
        to_port     = 0
    }
}

```

## » Example config to deny all traffic to any Subnet in the Default Network ACL:

This config denies all traffic in the Default ACL. This can be useful if you want a locked down default to force all resources in the VPC to assign a non-default ACL.

```
resource "aws_vpc" "mainvpc" {
  cidr_block = "10.1.0.0/16"
}

resource "aws_default_network_acl" "default" {
  default_network_acl_id = "${aws_vpc.mainvpc.default_network_acl_id}"

  # no rules defined, deny all traffic in this ACL
}
```

## » Argument Reference

The following arguments are supported:

- **default\_network\_acl\_id** - (Required) The Network ACL ID to manage. This attribute is exported from `aws_vpc`, or manually found via the AWS Console.
- **subnet\_ids** - (Optional) A list of Subnet IDs to apply the ACL to. See the notes below on managing Subnets in the Default Network ACL
- **ingress** - (Optional) Specifies an ingress rule. Parameters defined below.
- **egress** - (Optional) Specifies an egress rule. Parameters defined below.
- **tags** - (Optional) A mapping of tags to assign to the resource.

Both **egress** and **ingress** support the following keys:

- **from\_port** - (Required) The from port to match.
- **to\_port** - (Required) The to port to match.
- **rule\_no** - (Required) The rule number. Used for ordering.
- **action** - (Required) The action to take.
- **protocol** - (Required) The protocol to match. If using the -1 'all' protocol, you must specify a from and to port of 0.
- **cidr\_block** - (Optional) The CIDR block to match. This must be a valid network mask.
- **ipv6\_cidr\_block** - (Optional) The IPv6 CIDR block.
- **icmp\_type** - (Optional) The ICMP type to be used. Default 0.
- **icmp\_code** - (Optional) The ICMP type code to be used. Default 0.

Note: For more information on ICMP types and codes, see here: <http://www.nthelp.com/icmp.html>

## » Managing Subnets in the Default Network ACL

Within a VPC, all Subnets must be associated with a Network ACL. In order to "delete" the association between a Subnet and a non-default Network ACL, the association is destroyed by replacing it with an association between the Subnet and the Default ACL instead.

When managing the Default Network ACL, you cannot "remove" Subnets. Instead, they must be reassigned to another Network ACL, or the Subnet itself must be destroyed. Because of these requirements, removing the `subnet_ids` attribute from the configuration of a `aws_default_network_acl` resource may result in a reoccurring plan, until the Subnets are reassigned to another Network ACL or are destroyed.

Because Subnets are by default associated with the Default Network ACL, any non-explicit association will show up as a plan to remove the Subnet. For example: if you have a custom `aws_network_acl` with two subnets attached, and you remove the `aws_network_acl` resource, after successfully destroying this resource future plans will show a diff on the managed `aws_default_network_acl`, as those two Subnets have been orphaned by the now destroyed network acl and thus adopted by the Default Network ACL. In order to avoid a reoccurring plan, they will need to be reassigned, destroyed, or added to the `subnet_ids` attribute of the `aws_default_network_acl` entry.

## » Removing `aws_default_network_acl` from your configuration

Each AWS VPC comes with a Default Network ACL that cannot be deleted. The `aws_default_network_acl` allows you to manage this Network ACL, but Terraform cannot destroy it. Removing this resource from your configuration will remove it from your statefile and management, **but will not destroy the Network ACL**. All Subnets associations and ingress or egress rules will be left as they are at the time of removal. You can resume managing them via the AWS Console.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the Default Network ACL
- `vpc_id` - The ID of the associated VPC
- `ingress` - Set of ingress rules
- `egress` - Set of egress rules
- `subnet_ids` - IDs of associated Subnets

## » `aws__default__route__table`

Provides a resource to manage a Default VPC Routing Table.

Each VPC created in AWS comes with a Default Route Table that can be managed, but not destroyed. **This is an advanced resource**, and has special caveats to be aware of when using it. Please read this document in its entirety before using this resource. It is recommended you **do not** use both `aws_default_route_table` to manage the default route table **and** use the `aws_main_route_table_association`, due to possible conflict in routes.

The `aws_default_route_table` behaves differently from normal resources, in that Terraform does not *create* this resource, but instead attempts to "adopt" it into management. We can do this because each VPC created has a Default Route Table that cannot be destroyed, and is created with a single route.

When Terraform first adopts the Default Route Table, it **immediately removes all defined routes**. It then proceeds to create any routes specified in the configuration. This step is required so that only the routes specified in the configuration present in the Default Route Table.

For more information about Route Tables, see the AWS Documentation on Route Tables.

For more information about managing normal Route Tables in Terraform, see our documentation on `aws_route_table`.

**NOTE on Route Tables and Routes:** Terraform currently provides both a standalone Route resource and a Route Table resource with routes defined in-line. At this time you cannot use a Route Table with in-line routes in conjunction with any Route resources. Doing so will cause a conflict of rule settings and will overwrite routes.

### » Example usage with tags:

```
resource "aws_default_route_table" "r" {
  default_route_table_id = "${aws_vpc.foo.default_route_table_id}"

  route {
    # ...
  }

  tags {
    Name = "default table"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `default_route_table_id` - (Required) The ID of the Default Routing Table.
- `route` - (Optional) A list of route objects. Their keys are documented below.
- `tags` - (Optional) A mapping of tags to assign to the resource.
- `propagating_vgws` - (Optional) A list of virtual gateways for propagation.

Each route supports the following:

- `cidr_block` - (Required) The CIDR block of the route.
- `ipv6_cidr_block` - (Optional) The Ipv6 CIDR block of the route
- `egress_only_gateway_id` - (Optional) The Egress Only Internet Gateway ID.
- `gateway_id` - (Optional) The Internet Gateway ID.
- `nat_gateway_id` - (Optional) The NAT Gateway ID.
- `instance_id` - (Optional) The EC2 instance ID.
- `vpc_peering_connection_id` - (Optional) The VPC Peering ID.
- `network_interface_id` - (Optional) The ID of the elastic network interface (eni) to use.

Each route must contain either a `gateway_id`, an `instance_id`, a `nat_gateway_id`, a `vpc_peering_connection_id` or a `network_interface_id`. Note that the default route, mapping the VPC's CIDR block to "local", is created implicitly and cannot be specified.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the routing table

## » `aws_default_security_group`

Provides a resource to manage the default AWS Security Group.

For EC2 Classic accounts, each region comes with a Default Security Group. Additionally, each VPC created in AWS comes with a Default Security Group that can be managed, but not destroyed. **This is an advanced resource**, and has special caveats to be aware of when using it. Please read this document in its entirety before using this resource.

The `aws_default_security_group` behaves differently from normal resources, in that Terraform does not *create* this resource, but instead "adopts" it into

management. We can do this because these default security groups cannot be destroyed, and are created with a known set of default ingress/egress rules.

When Terraform first adopts the Default Security Group, it **immediately removes all ingress and egress rules in the Security Group**. It then proceeds to create any rules specified in the configuration. This step is required so that only the rules specified in the configuration are created.

This resource treats its inline rules as absolute; only the rules defined inline are created, and any additions/removals external to this resource will result in diff shown. For these reasons, this resource is incompatible with the `aws_security_group_rule` resource.

For more information about Default Security Groups, see the AWS Documentation on Default Security Groups.

## » Basic Example Usage, with default rules

The following config gives the Default Security Group the same rules that AWS provides by default, but pulls the resource under management by Terraform. This means that any ingress or egress rules added or changed will be detected as drift.

```
resource "aws_vpc" "mainvpc" {
  cidr_block = "10.1.0.0/16"
}

resource "aws_default_security_group" "default" {
  vpc_id = "${aws_vpc.mainvpc.id}"

  ingress {
    protocol = -1
    self     = true
    from_port = 0
    to_port   = 0
  }

  egress {
    from_port = 0
    to_port   = 0
    protocol  = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }
}
```

## » Example config to deny all Egress traffic, allowing Ingress

The following denies all Egress traffic by omitting any `egress` rules, while including the default `ingress` rule to allow all traffic.

```
resource "aws_vpc" "mainvpc" {
  cidr_block = "10.1.0.0/16"
}

resource "aws_default_security_group" "default" {
  vpc_id = "${aws_vpc.mainvpc.id}"

  ingress {
    protocol = -1
    self     = true
    from_port = 0
    to_port  = 0
  }
}
```

## » Argument Reference

The arguments of an `aws_default_security_group` differ slightly from `aws_security_group` resources. Namely, the `name` argument is computed, and the `name_prefix` attribute removed. The following arguments are still supported:

- `ingress` - (Optional) Can be specified multiple times for each ingress rule. Each ingress block supports fields documented below.
- `egress` - (Optional, VPC only) Can be specified multiple times for each egress rule. Each egress block supports fields documented below.
- `vpc_id` - (Optional, Forces new resource) The VPC ID. **Note that changing the `vpc_id` will *not* restore any default security group rules that were modified, added, or removed.** It will be left in its current state
- `tags` - (Optional) A mapping of tags to assign to the resource.

## » Usage

With the exceptions mentioned above, `aws_default_security_group` should have identical behavior to `aws_security_group`. Please consult `AWS_SECURITY_GROUP` for further usage documentation.



## » Removing `aws_default_security_group` from your configuration

Each AWS VPC (or region, if using EC2 Classic) comes with a Default Security Group that cannot be deleted. The `aws_default_security_group` allows you to manage this Security Group, but Terraform cannot destroy it. Removing this resource from your configuration will remove it from your statefile and management, but will not destroy the Security Group. All ingress or egress rules will be left as they are at the time of removal. You can resume managing them via the AWS Console.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the security group
- `vpc_id` - The VPC ID.
- `owner_id` - The owner ID.
- `name` - The name of the security group
- `description` - The description of the security group
- `ingress` - The ingress rules. See above for more.
- `egress` - The egress rules. See above for more.

## » `aws_default_subnet`

Provides a resource to manage a default AWS VPC subnet in the current region.

The `aws_default_subnet` behaves differently from normal resources, in that Terraform does not *create* this resource, but instead "adopts" it into management.

## » Example Usage

Basic usage with tags:

```
resource "aws_default_subnet" "default_az1" {
  availability_zone = "us-west-2a"

  tags {
    Name = "Default subnet for us-west-2a"
  }
}
```

## » Argument Reference

The arguments of an `aws_default_subnet` differ from `aws_subnet` resources. Namely, the `availability_zone` argument is required and the `vpc_id`, `cidr_block`, `ipv6_cidr_block`, and `assign_ipv6_address_on_creation` arguments are computed. The following arguments are still supported:

- `map_public_ip_on_launch` - (Optional) Specify true to indicate that instances launched into the subnet should be assigned a public IP address.
- `tags` - (Optional) A mapping of tags to assign to the resource.

## » Removing `aws_default_subnet` from your configuration

The `aws_default_subnet` resource allows you to manage a region's default VPC subnet, but Terraform cannot destroy it. Removing this resource from your configuration will remove it from your statefile and management, but will not destroy the subnet. You can resume managing the subnet via the AWS Console.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the subnet
- `availability_zone` - The AZ for the subnet.
- `cidr_block` - The CIDR block for the subnet.
- `vpc_id` - The VPC ID.
- `ipv6_association_id` - The association ID for the IPv6 CIDR block.
- `ipv6_cidr_block` - The IPv6 CIDR block.

## » `aws__default__vpc`

Provides a resource to manage the default AWS VPC in the current region.

For AWS accounts created after 2013-12-04, each region comes with a Default VPC. **This is an advanced resource**, and has special caveats to be aware of when using it. Please read this document in its entirety before using this resource.

The `aws_default_vpc` behaves differently from normal resources, in that Terraform does not *create* this resource, but instead "adopts" it into management.

## » Example Usage

Basic usage with tags:

```
resource "aws_default_vpc" "default" {
  tags {
    Name = "Default VPC"
  }
}
```

## » Argument Reference

The arguments of an `aws_default_vpc` differ slightly from `aws_vpc` resources. Namely, the `cidr_block`, `instance_tenancy` and `assign_generated_ipv6_cidr_block` arguments are computed. The following arguments are still supported:

- `enable_dns_support` - (Optional) A boolean flag to enable/disable DNS support in the VPC. Defaults true.
- `enable_dns_hostnames` - (Optional) A boolean flag to enable/disable DNS hostnames in the VPC. Defaults false.
- `enable_classiclink` - (Optional) A boolean flag to enable/disable ClassicLink for the VPC. Only valid in regions and accounts that support EC2 Classic. See the ClassicLink documentation for more information. Defaults false.
- `tags` - (Optional) A mapping of tags to assign to the resource.

## » Removing `aws_default_vpc` from your configuration

The `aws_default_vpc` resource allows you to manage a region's default VPC, but Terraform cannot destroy it. Removing this resource from your configuration will remove it from your statefile and management, but will not destroy the VPC. You can resume managing the VPC via the AWS Console.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the VPC
- `cidr_block` - The CIDR block of the VPC
- `instance_tenancy` - Tenancy of instances spin up within VPC.
- `enable_dns_support` - Whether or not the VPC has DNS support
- `enable_dns_hostnames` - Whether or not the VPC has DNS hostname support
- `enable_classiclink` - Whether or not the VPC has Classiclink enabled
- `assign_generated_ipv6_cidr_block` - Whether or not an Amazon-provided IPv6 CIDR block with a /56 prefix length for the VPC was assigned

- `main_route_table_id` - The ID of the main route table associated with this VPC. Note that you can change a VPC's main route table by using an `aws_main_route_table_association`
- `default_network_acl_id` - The ID of the network ACL created by default on VPC creation
- `default_security_group_id` - The ID of the security group created by default on VPC creation
- `default_route_table_id` - The ID of the route table created by default on VPC creation
- `ipv6_association_id` - The association ID for the IPv6 CIDR block of the VPC
- `ipv6_cidr_block` - The IPv6 CIDR block of the VPC

## » `aws__default__vpc__dhcp__options`

Provides a resource to manage the default AWS DHCP Options Set in the current region.

Each AWS region comes with a default set of DHCP options. **This is an advanced resource**, and has special caveats to be aware of when using it. Please read this document in its entirety before using this resource.

The `aws_default_vpc_dhcp_options` behaves differently from normal resources, in that Terraform does not *create* this resource, but instead "adopts" it into management.

## » Example Usage

Basic usage with tags:

```
resource "aws_default_vpc_dhcp_options" "default" {
  tags {
    Name = "Default DHCP Option Set"
  }
}
```

## » Argument Reference

The arguments of an `aws_default_vpc_dhcp_options` differ slightly from `aws_vpc_dhcp_options` resources. Namely, the `domain_name`, `domain_name_servers` and `ntp_servers` arguments are computed. The following arguments are still supported:

- `netbios_name_servers` - (Optional) List of NETBIOS name servers.

- **netbios\_node\_type** - (Optional) The NetBIOS node type (1, 2, 4, or 8). AWS recommends to specify 2 since broadcast and multicast are not supported in their network. For more information about these node types, see RFC 2132.
- **tags** - (Optional) A mapping of tags to assign to the resource.

#### » Removing `aws_default_vpc_dhcp_options` from your configuration

The `aws_default_vpc_dhcp_options` resource allows you to manage a region's default DHCP Options Set, but Terraform cannot destroy it. Removing this resource from your configuration will remove it from your statefile and management, but will not destroy the DHCP Options Set. You can resume managing the DHCP Options Set via the AWS Console.

#### » Attributes Reference

The following attributes are exported:

- **id** - The ID of the DHCP Options Set.

#### » `aws_egress_only_internet_gateway`

[IPv6 only] Creates an egress-only Internet gateway for your VPC. An egress-only Internet gateway is used to enable outbound communication over IPv6 from instances in your VPC to the Internet, and prevents hosts outside of your VPC from initiating an IPv6 connection with your instance.

#### » Example Usage

```
resource "aws_vpc" "foo" {
  cidr_block = "10.1.0.0/16"
  assign_generated_ipv6_cidr_block = true
}

resource "aws_egress_only_internet_gateway" "foo" {
  vpc_id = "${aws_vpc.foo.id}"
}
```

#### » Argument Reference

The following arguments are supported:

- `vpc_id` - (Required) The VPC ID to create in.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the Egress Only Internet Gateway.

## » `aws_flow_log`

Provides a VPC/Subnet/ENI Flow Log to capture IP traffic for a specific network interface, subnet, or VPC. Logs are sent to a CloudWatch Log Group.

## » Example Usage

```
resource "aws_flow_log" "test_flow_log" {
  log_group_name = "${aws_cloudwatch_log_group.test_log_group.name}"
  iam_role_arn   = "${aws_iam_role.test_role.arn}"
  vpc_id         = "${aws_vpc.default.id}"
  traffic_type   = "ALL"
}

resource "aws_cloudwatch_log_group" "test_log_group" {
  name = "test_log_group"
}

resource "aws_iam_role" "test_role" {
  name = "test_role"

  assume_role_policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "vpc-flow-logs.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

```

EOF
}

resource "aws_iam_role_policy" "test_policy" {
  name = "test_policy"
  role = "${aws_iam_role.test_role.id}"

  policy = <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:DescribeLogGroups",
        "logs:DescribeLogStreams"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
EOF
}

```

## » Argument Reference

The following arguments are supported:

- `log_group_name` - (Required) The name of the CloudWatch log group
- `iam_role_arn` - (Required) The ARN for the IAM role that's used to post flow logs to a CloudWatch Logs log group
- `vpc_id` - (Optional) VPC ID to attach to
- `subnet_id` - (Optional) Subnet ID to attach to
- `eni_id` - (Optional) Elastic Network Interface ID to attach to
- `traffic_type` - (Required) The type of traffic to capture. Valid values: ACCEPT, REJECT, ALL

## » Attributes Reference

The following attributes are exported:

- `id` - The Flow Log ID

## » Import

Flow Logs can be imported using the id, e.g.

```
$ terraform import aws_flow_log.test_flow_log fl-1a2b3c4d
```

## » aws\_\_internet\_\_gateway

Provides a resource to create a VPC Internet Gateway.

## » Example Usage

```
resource "aws_internet_gateway" "gw" {
  vpc_id = "${aws_vpc.main.id}"

  tags {
    Name = "main"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `vpc_id` - (Required) The VPC ID to create in.
- `tags` - (Optional) A mapping of tags to assign to the resource.

**Note:** It's recommended to denote that the AWS Instance or Elastic IP depends on the Internet Gateway. For example:

```
resource "aws_internet_gateway" "gw" {
  vpc_id = "${aws_vpc.main.id}"
}

resource "aws_instance" "foo" {
  depends_on = ["aws_internet_gateway.gw"]
}
```

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the Internet Gateway.



## » Import

Internet Gateways can be imported using the `id`, e.g.

```
$ terraform import aws_internet_gateway.gw igw-c0a643a9
```

## » `aws_main_route_table_association`

Provides a resource for managing the main routing table of a VPC.

## » Example Usage

```
resource "aws_main_route_table_association" "a" {  
    vpc_id          = "${aws_vpc.foo.id}"  
    route_table_id = "${aws_route_table.bar.id}"  
}
```

## » Argument Reference

The following arguments are supported:

- `vpc_id` - (Required) The ID of the VPC whose main route table should be set
- `route_table_id` - (Required) The ID of the Route Table to set as the new main route table for the target VPC

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the Route Table Association
- `original_route_table_id` - Used internally, see **Notes** below

## » Notes

On VPC creation, the AWS API always creates an initial Main Route Table. This resource records the ID of that Route Table under `original_route_table_id`. The "Delete" action for a `main_route_table_association` consists of resetting this original table as the Main Route Table for the VPC. You'll see this additional Route Table in the AWS console; it must remain intact in order for the `main_route_table_association` delete to work properly.

## » `aws_nat_gateway`

Provides a resource to create a VPC NAT Gateway.

### » Example Usage

```
resource "aws_nat_gateway" "gw" {
  allocation_id = "${aws_eip.nat.id}"
  subnet_id     = "${aws_subnet.public.id}"
}
```

Usage with tags:

```
resource "aws_nat_gateway" "gw" {
  allocation_id = "${aws_eip.nat.id}"
  subnet_id     = "${aws_subnet.public.id}"

  tags {
    Name = "gw NAT"
  }
}
```

### » Argument Reference

The following arguments are supported:

- `allocation_id` - (Required) The Allocation ID of the Elastic IP address for the gateway.
- `subnet_id` - (Required) The Subnet ID of the subnet in which to place the gateway.
- `tags` - (Optional) A mapping of tags to assign to the resource.

**Note:** It's recommended to denote that the NAT Gateway depends on the Internet Gateway for the VPC in which the NAT Gateway's subnet is located. For example:

```
resource "aws_internet_gateway" "gw" {
  vpc_id = "${aws_vpc.main.id}"
}

resource "aws_nat_gateway" "gw" {
  //other arguments

  depends_on = ["aws_internet_gateway.gw"]
}
```

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the NAT Gateway.
- `allocation_id` - The Allocation ID of the Elastic IP address for the gateway.
- `subnet_id` - The Subnet ID of the subnet in which the NAT gateway is placed.
- `network_interface_id` - The ENI ID of the network interface created by the NAT gateway.
- `private_ip` - The private IP address of the NAT Gateway.
- `public_ip` - The public IP address of the NAT Gateway.

## » Import

NAT Gateways can be imported using the `id`, e.g.

```
$ terraform import aws_nat_gateway.private_gw nat-05dba92075d71c408
```

## » `aws_network_acl`

Provides an network ACL resource. You might set up network ACLs with rules similar to your security groups in order to add an additional layer of security to your VPC.

**NOTE on Network ACLs and Network ACL Rules:** Terraform currently provides both a standalone Network ACL Rule resource and a Network ACL resource with rules defined in-line. At this time you cannot use a Network ACL with in-line rules in conjunction with any Network ACL Rule resources. Doing so will cause a conflict of rule settings and will overwrite rules.

## » Example Usage

```
resource "aws_network_acl" "main" {
  vpc_id = "${aws_vpc.main.id}"

  egress {
    protocol    = "tcp"
    rule_no     = 200
    action      = "allow"
    cidr_block  = "10.3.0.0/18"
    from_port   = 443
    to_port     = 443
  }
}
```

```

}

ingress {
    protocol    = "tcp"
    rule_no     = 100
    action      = "allow"
    cidr_block  = "10.3.0.0/18"
    from_port   = 80
    to_port     = 80
}

tags {
    Name = "main"
}
}

```

## » Argument Reference

The following arguments are supported:

- **vpc\_id** - (Required) The ID of the associated VPC.
- **subnet\_ids** - (Optional) A list of Subnet IDs to apply the ACL to
- **subnet\_id** - (Optional, Deprecated) The ID of the associated Subnet. This attribute is deprecated, please use the **subnet\_ids** attribute instead
- **ingress** - (Optional) Specifies an ingress rule. Parameters defined below.
- **egress** - (Optional) Specifies an egress rule. Parameters defined below.
- **tags** - (Optional) A mapping of tags to assign to the resource.

Both **egress** and **ingress** support the following keys:

- **from\_port** - (Required) The from port to match.
- **to\_port** - (Required) The to port to match.
- **rule\_no** - (Required) The rule number. Used for ordering.
- **action** - (Required) The action to take.
- **protocol** - (Required) The protocol to match. If using the -1 'all' protocol, you must specify a from and to port of 0.
- **cidr\_block** - (Optional) The CIDR block to match. This must be a valid network mask.
- **ipv6\_cidr\_block** - (Optional) The IPv6 CIDR block.
- **icmp\_type** - (Optional) The ICMP type to be used. Default 0.
- **icmp\_code** - (Optional) The ICMP type code to be used. Default 0.

Note: For more information on ICMP types and codes, see here: <http://www.nthelp.com/icmp.html>

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the network ACL

## » Import

Network ACLs can be imported using the `id`, e.g.

```
$ terraform import aws_network_acl.main acl-7aaabd18
```

## » `aws_network_acl_rule`

Creates an entry (a rule) in a network ACL with the specified rule number.

**NOTE on Network ACLs and Network ACL Rules:** Terraform currently provides both a standalone Network ACL Rule resource and a Network ACL resource with rules defined in-line. At this time you cannot use a Network ACL with in-line rules in conjunction with any Network ACL Rule resources. Doing so will cause a conflict of rule settings and will overwrite rules.

## » Example Usage

```
resource "aws_network_acl" "bar" {
  vpc_id = "${aws_vpc.foo.id}"
}

resource "aws_network_acl_rule" "bar" {
  network_acl_id = "${aws_network_acl.bar.id}"
  rule_number    = 200
  egress         = false
  protocol       = "tcp"
  rule_action    = "allow"
  cidr_block     = "0.0.0.0/0"
  from_port      = 22
  to_port        = 22
}
```

**Note:** One of either `cidr_block` or `ipv6_cidr_block` is required.

## » Argument Reference

The following arguments are supported:

- **network\_acl\_id** - (Required) The ID of the network ACL.
- **rule\_number** - (Required) The rule number for the entry (for example, 100). ACL entries are processed in ascending order by rule number.
- **egress** - (Optional, bool) Indicates whether this is an egress rule (rule is applied to traffic leaving the subnet). Default **false**.
- **protocol** - (Required) The protocol. A value of -1 means all protocols.
- **rule\_action** - (Required) Indicates whether to allow or deny the traffic that matches the rule. Accepted values: **allow** | **deny**
- **cidr\_block** - (Optional) The network range to allow or deny, in CIDR notation (for example 172.16.0.0/24 ).
- **ipv6\_cidr\_block** - (Optional) The IPv6 CIDR block to allow or deny.
- **from\_port** - (Optional) The from port to match.
- **to\_port** - (Optional) The to port to match.
- **icmp\_type** - (Optional) ICMP protocol: The ICMP type. Required if specifying ICMP for the protocol. e.g. -1
- **icmp\_code** - (Optional) ICMP protocol: The ICMP code. Required if specifying ICMP for the protocol. e.g. -1

**NOTE:** If the value of **protocol** is -1 or **all**, the **from\_port** and **to\_port** values will be ignored and the rule will apply to all ports.

**NOTE:** If the value of **icmp\_type** is -1 (which results in a wildcard ICMP type), the **icmp\_code** must also be set to -1 (wildcard ICMP code).

Note: For more information on ICMP types and codes, see here: <http://www.nthelp.com/icmp.html>

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the network ACL Rule

## » aws\_\_network\_\_interface

Provides an Elastic network interface (ENI) resource.

## » Example Usage

```
resource "aws_network_interface" "test" {
  subnet_id      = "${aws_subnet.public_a.id}"
```

```

private_ips      = ["10.0.0.50"]
security_groups = ["${aws_security_group.web.id}"]

attachment {
    instance      = "${aws_instance.test.id}"
    device_index = 1
}
}

```

## » Argument Reference

The following arguments are supported:

- **subnet\_id** - (Required) Subnet ID to create the ENI in.
- **description** - (Optional) A description for the network interface.
- **private\_ips** - (Optional) List of private IPs to assign to the ENI.
- **private\_ips\_count** - (Optional) Number of private IPs to assign to the ENI.
- **security\_groups** - (Optional) List of security group IDs to assign to the ENI.
- **attachment** - (Optional) Block to define the attachment of the ENI. Documented below.
- **source\_dest\_check** - (Optional) Whether to enable source destination checking for the ENI. Default true.
- **tags** - (Optional) A mapping of tags to assign to the resource.

The **attachment** block supports:

- **instance** - (Required) ID of the instance to attach to.
- **device\_index** - (Required) Integer to define the devices index.

## » Attributes Reference

The following attributes are exported:

- **subnet\_id** - Subnet ID the ENI is in.
- **description** - A description for the network interface.
- **private\_ips** - List of private IPs assigned to the ENI.
- **security\_groups** - List of security groups attached to the ENI.
- **attachment** - Block defining the attachment of the ENI.
- **source\_dest\_check** - Whether source destination checking is enabled
- **tags** - Tags assigned to the ENI.

## » Import

Network Interfaces can be imported using the `id`, e.g.

```
$ terraform import aws_network_interface.test eni-e5aa89a3
```

## » `aws__network__interface__attachment`

Attach an Elastic network interface (ENI) resource with EC2 instance.

## » Example Usage

```
resource "aws_network_interface_attachment" "test" {
  instance_id = "${aws_instance.test.id}"
  network_interface_id = "${aws_network_interface.test.id}"
  device_index = 0
}
```

## » Argument Reference

The following arguments are supported:

- `instance_id` - (Required) Instance ID to attach.
- `network_interface_id` - (Required) ENI ID to attach.
- `device_index` - (Required) Network interface index (int).

## » Attributes Reference

The following attributes are exported:

- `instance_id` - Instance ID.
- `network_interface_id` - Network interface ID.
- `attachment_id` - The ENI Attachment ID.
- `status` - The status of the Network Interface Attachment.

## » `aws__route`

Provides a resource to create a routing table entry (a route) in a VPC routing table.



**NOTE on Route Tables and Routes:** Terraform currently provides both a standalone Route resource and a Route Table resource with routes defined in-line. At this time you cannot use a Route Table with in-line routes in conjunction with any Route resources. Doing so will cause a conflict of rule settings and will overwrite rules.

» **Example usage:**

```
resource "aws_route" "r" {
  route_table_id      = "rtb-4fbb3ac4"
  destination_cidr_block = "10.0.1.0/22"
  vpc_peering_connection_id = "pcx-45ff3dc1"
  depends_on          = ["aws_route_table.testing"]
}
```

» **Example IPv6 Usage:**

```
resource "aws_vpc" "vpc" {
  cidr_block = "10.1.0.0/16"
  assign_generated_ipv6_cidr_block = true
}

resource "aws_egress_only_internet_gateway" "egress" {
  vpc_id = "${aws_vpc.vpc.id}"
}

resource "aws_route" "r" {
  route_table_id      = "rtb-4fbb3ac4"
  destination_ipv6_cidr_block = "::/0"
  egress_only_gateway_id = "${aws_egress_only_internet_gateway.egress.id}"
}
```

» **Argument Reference**

The following arguments are supported:

- `route_table_id` - (Required) The ID of the routing table.
- `destination_cidr_block` - (Optional) The destination CIDR block.
- `destination_ipv6_cidr_block` - (Optional) The destination IPv6 CIDR block.
- `vpc_peering_connection_id` - (Optional) An ID of a VPC peering connection.
- `egress_only_gateway_id` - (Optional) An ID of a VPC Egress Only Internet Gateway.

- `gateway_id` - (Optional) An ID of a VPC internet gateway or a virtual private gateway.
- `nat_gateway_id` - (Optional) An ID of a VPC NAT gateway.
- `instance_id` - (Optional) An ID of an EC2 instance.
- `network_interface_id` - (Optional) An ID of a network interface.

Each route must contain either a `gateway_id`, `egress_only_gateway_id`, `nat_gateway_id`, an `instance_id` or a `vpc_peering_connection_id` or a `network_interface_id`. Note that the default route, mapping the VPC's CIDR block to "local", is created implicitly and cannot be specified.

## » Attributes Reference

The following attributes are exported:

**NOTE:** Only the target type that is specified (one of the above) will be exported as an attribute once the resource is created.

- `route_table_id` - The ID of the routing table.
- `destination_cidr_block` - The destination CIDR block.
- `destination_ipv6_cidr_block` - The destination IPv6 CIDR block.
- `vpc_peering_connection_id` - An ID of a VPC peering connection.
- `egress_only_gateway_id` - An ID of a VPC Egress Only Internet Gateway.
- `gateway_id` - An ID of a VPC internet gateway or a virtual private gateway.
- `nat_gateway_id` - An ID of a VPC NAT gateway.
- `instance_id` - An ID of a NAT instance.
- `network_interface_id` - An ID of a network interface.

## » Timeouts

`aws_route` provides the following Timeouts configuration options:

- `create` - (Default 2 minutes) Used for route creation
- `delete` - (Default 5 minutes) Used for route deletion

## » `aws_route_table`

Provides a resource to create a VPC routing table.

**NOTE on Route Tables and Routes:** Terraform currently provides both a standalone Route resource and a Route Table resource with routes defined in-line. At this time you cannot use a Route Table with in-line routes in conjunction

with any Route resources. Doing so will cause a conflict of rule settings and will overwrite rules.

**NOTE on `gateway_id` and `nat_gateway_id`:** The AWS API is very forgiving with these two attributes and the `aws_route_table` resource can be created with a NAT ID specified as a Gateway ID attribute. This *will* lead to a permanent diff between your configuration and statefile, as the API returns the correct parameters in the returned route table. If you're experiencing constant diffs in your `aws_route_table` resources, the first thing to check is whether or not you're specifying a NAT ID instead of a Gateway ID, or vice-versa.

**NOTE on `propagating_vgws` and the `aws_vpn_gateway_route_propagation` resource:** If the `propagating_vgws` argument is present, it's not supported to *also* define route propagations using `aws_vpn_gateway_route_propagation`, since this resource will delete any propagating gateways not explicitly listed in `propagating_vgws`. Omit this argument when defining route propagation using the separate resource.

## » Example usage with tags:

```
resource "aws_route_table" "r" {
  vpc_id = "${aws_vpc.default.id}"

  route {
    cidr_block = "10.0.1.0/24"
    gateway_id = "${aws_internet_gateway.main.id}"
  }

  route {
    ipv6_cidr_block = ":::/0"
    egress_only_gateway_id = "${aws_egress_only_internet_gateway.foo.id}"
  }

  tags {
    Name = "main"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `vpc_id` - (Required) The VPC ID.
- `route` - (Optional) A list of route objects. Their keys are documented below.

- `tags` - (Optional) A mapping of tags to assign to the resource.
- `propagating_vgws` - (Optional) A list of virtual gateways for propagation.

Each route supports the following:

- `cidr_block` - (Optional) The CIDR block of the route.
- `ipv6_cidr_block` - (Optional) The Ipv6 CIDR block of the route
- `egress_only_gateway_id` - (Optional) The Egress Only Internet Gateway ID.
- `gateway_id` - (Optional) The Internet Gateway ID.
- `nat_gateway_id` - (Optional) The NAT Gateway ID.
- `instance_id` - (Optional) The EC2 instance ID.
- `vpc_peering_connection_id` - (Optional) The VPC Peering ID.
- `network_interface_id` - (Optional) The ID of the elastic network interface (eni) to use.

Each route must contain either a `gateway_id`, an `instance_id`, a `nat_gateway_id`, a `vpc_peering_connection_id` or a `network_interface_id`. Note that the default route, mapping the VPC's CIDR block to "local", is created implicitly and cannot be specified.

## » Attributes Reference

The following attributes are exported: ~> **NOTE:** Only the target that is entered is exported as a readable attribute once the route resource is created.

- `id` - The ID of the routing table

## » Import

Route Tables can be imported using the `route table id`, e.g.

```
$ terraform import aws_route_table.public_rt rtb-22574640
```

## » `aws_route_table_association`

Provides a resource to create an association between a subnet and routing table.

## » Example Usage

```
resource "aws_route_table_association" "a" {
  subnet_id      = "${aws_subnet.foo.id}"
  route_table_id = "${aws_route_table.bar.id}"
}
```

## » Argument Reference

The following arguments are supported:

- `subnet_id` - (Required) The subnet ID to create an association.
- `route_table_id` - (Required) The ID of the routing table to associate with.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the association

## » `aws_security_group`

Provides a security group resource.

**NOTE on Security Groups and Security Group Rules:** Terraform currently provides both a standalone Security Group Rule resource (a single `ingress` or `egress` rule), and a Security Group resource with `ingress` and `egress` rules defined in-line. At this time you cannot use a Security Group with in-line rules in conjunction with any Security Group Rule resources. Doing so will cause a conflict of rule settings and will overwrite rules.

## » Example Usage

Basic usage

```
resource "aws_security_group" "allow_all" {
  name           = "allow_all"
  description    = "Allow all inbound traffic"
  vpc_id         = "${aws_vpc.main.id}"

  ingress {
    from_port     = 0
    to_port       = 0
    protocol      = "-1"
    cidr_blocks   = ["0.0.0.0/0"]
  }

  egress {
    from_port     = 0
    to_port       = 0
  }
```

```

        protocol      = "-1"
        cidr_blocks    = ["0.0.0.0/0"]
        prefix_list_ids = ["pl-12c4e678"]
    }
}

```

Basic usage with tags:

```

resource "aws_security_group" "allow_all" {
    name      = "allow_all"
    description = "Allow all inbound traffic"

    ingress {
        from_port    = 0
        to_port      = 65535
        protocol      = "tcp"
        cidr_blocks = ["0.0.0.0/0"]
    }

    tags {
        Name = "allow_all"
    }
}

```

## » Argument Reference

The following arguments are supported:

- **name** - (Optional, Forces new resource) The name of the security group. If omitted, Terraform will assign a random, unique name
- **name\_prefix** - (Optional, Forces new resource) Creates a unique name beginning with the specified prefix. Conflicts with **name**.
- **description** - (Optional, Forces new resource) The security group description. Defaults to "Managed by Terraform". Cannot be "". **NOTE:** This field maps to the AWS `GroupDescription` attribute, for which there is no Update API. If you'd like to classify your security groups in a way that can be updated, use **tags**.
- **ingress** - (Optional) Can be specified multiple times for each ingress rule. Each ingress block supports fields documented below.
- **egress** - (Optional, VPC only) Can be specified multiple times for each egress rule. Each egress block supports fields documented below.
- **revoke\_rules\_on\_delete** - (Optional) Instruct Terraform to revoke all of the Security Groups attached ingress and egress rules before deleting the rule itself. This is normally not needed, however certain AWS services such as Elastic Map Reduce may automatically add required rules to security groups used with the service, and those rules may contain a

cyclic dependency that prevent the security groups from being destroyed without removing the dependency first. Default **false**

- **vpc\_id** - (Optional, Forces new resource) The VPC ID.
- **tags** - (Optional) A mapping of tags to assign to the resource.

The **ingress** block supports:

- **cidr\_blocks** - (Optional) List of CIDR blocks.
- **ipv6\_cidr\_blocks** - (Optional) List of IPv6 CIDR blocks.
- **from\_port** - (Required) The start port (or ICMP type number if protocol is "icmp")
- **protocol** - (Required) The protocol. If you select a protocol of "-1" (semantically equivalent to "all", which is not a valid value here), you must specify a "from\_port" and "to\_port" equal to 0. If not icmp, tcp, udp, or "-1" use the protocol number
- **security\_groups** - (Optional) List of security group Group Names if using EC2-Classic, or Group IDs if using a VPC.
- **self** - (Optional) If true, the security group itself will be added as a source to this ingress rule.
- **to\_port** - (Required) The end range port (or ICMP code if protocol is "icmp").
- **description** - (Optional) Description of this ingress rule.

The **egress** block supports:

- **cidr\_blocks** - (Optional) List of CIDR blocks.
- **ipv6\_cidr\_blocks** - (Optional) List of IPv6 CIDR blocks.
- **prefix\_list\_ids** - (Optional) List of prefix list IDs (for allowing access to VPC endpoints)
- **from\_port** - (Required) The start port (or ICMP type number if protocol is "icmp")
- **protocol** - (Required) The protocol. If you select a protocol of "-1" (semantically equivalent to "all", which is not a valid value here), you must specify a "from\_port" and "to\_port" equal to 0. If not icmp, tcp, udp, or "-1" use the protocol number
- **security\_groups** - (Optional) List of security group Group Names if using EC2-Classic, or Group IDs if using a VPC.
- **self** - (Optional) If true, the security group itself will be added as a source to this egress rule.
- **to\_port** - (Required) The end range port (or ICMP code if protocol is "icmp").
- **description** - (Optional) Description of this egress rule.

**NOTE on Egress rules:** By default, AWS creates an **ALLOW ALL** egress rule when creating a new Security Group inside of a VPC. When creating a new Security Group inside a VPC, **Terraform will remove this default rule**, and require you specifically re-create it if you desire that rule. We feel this leads to fewer surprises in terms of controlling your egress rules. If you desire this

rule to be in place, you can use this **egress** block:

```
egress {
  from_port = 0
  to_port = 0
  protocol = "-1"
  cidr_blocks = ["0.0.0.0/0"]
}
```

## » Usage with prefix list IDs

Prefix list IDs are managed by AWS internally. Prefix list IDs are associated with a prefix list name, or service name, that is linked to a specific region. Prefix list IDs are exported on VPC Endpoints, so you can use this format:

```
# ...
egress {
  from_port = 0
  to_port = 0
  protocol = "-1"
  prefix_list_ids = ["${aws_vpc_endpoint.my_endpoint.prefix_list_id}"]
}
# ...
resource "aws_vpc_endpoint" "my_endpoint" {
  # ...
}
```

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the security group
- **arn** - The ARN of the security group
- **vpc\_id** - The VPC ID.
- **owner\_id** - The owner ID.
- **name** - The name of the security group
- **description** - The description of the security group
- **ingress** - The ingress rules. See above for more.
- **egress** - The egress rules. See above for more.

## » Timeouts

**aws\_security\_group** provides the following Timeouts configuration options:



- **create** - (Default 10 minutes) How long to wait for a security group to be created.
- **delete** - (Default 10 minutes) How long to wait for a security group to be deleted.

## » Import

Security Groups can be imported using the `security_group_id`, e.g.

```
$ terraform import aws_security_group.elb_sg sg-903004f8
```

## » `aws_network_interface_sg_attachment`

This resource attaches a security group to an Elastic Network Interface (ENI). It can be used to attach a security group to any existing ENI, be it a secondary ENI or one attached as the primary interface on an instance.

**NOTE on instances, interfaces, and security groups:** Terraform currently provides the capability to assign security groups via the `aws_instance` and the `aws_network_interface` resources. Using this resource in conjunction with security groups provided in-line in those resources will cause conflicts, and will lead to spurious diffs and undefined behavior - please use one or the other.

## » Example Usage

The following provides a very basic example of setting up an instance (provided by `instance`) in the default security group, creating a security group (provided by `sg`) and then attaching the security group to the instance's primary network interface via the `aws_network_interface_sg_attachment` resource, named `sg_attachment`:

```
data "aws_ami" "ami" {
  most_recent = true

  filter {
    name     = "name"
    values   = ["amzn-ami-hvm-*"]
  }

  owners = ["amazon"]
}

resource "aws_instance" "instance" {
  instance_type = "t2.micro"
```

```

ami          = "${data.aws_ami.ami.id}"

tags = {
  "type" = "terraform-test-instance"
}

resource "aws_security_group" "sg" {
  tags = {
    "type" = "terraform-test-security-group"
  }
}

resource "aws_network_interface_sg_attachment" "sg_attachment" {
  security_group_id = "${aws_security_group.sg.id}"
  network_interface_id = "${aws_instance.instance.primary_network_interface_id}"
}

```

In this example, `instance` is provided by the `aws_instance` data source, fetching an external instance, possibly not managed by Terraform. `sg_attachment` then attaches to the output instance's `network_interface_id`:

```

data "aws_instance" "instance" {
  instance_id = "i-1234567890abcdef0"
}

resource "aws_security_group" "sg" {
  tags = {
    "type" = "terraform-test-security-group"
  }
}

resource "aws_network_interface_sg_attachment" "sg_attachment" {
  security_group_id = "${aws_security_group.sg.id}"
  network_interface_id = "${data.aws_instance.instance.network_interface_id}"
}

```

## » Argument Reference

- `security_group_id` - (Required) The ID of the security group.
- `network_interface_id` - (Required) The ID of the network interface to attach to.

## » Output Reference

There are no outputs for this resource.

## » `aws_security_group_rule`

Provides a security group rule resource. Represents a single **ingress** or **egress** group rule, which can be added to external Security Groups.

**NOTE on Security Groups and Security Group Rules:** Terraform currently provides both a standalone Security Group Rule resource (a single **ingress** or **egress** rule), and a Security Group resource with **ingress** and **egress** rules defined in-line. At this time you cannot use a Security Group with in-line rules in conjunction with any Security Group Rule resources. Doing so will cause a conflict of rule settings and will overwrite rules.

## » Example Usage

Basic usage

```
resource "aws_security_group_rule" "allow_all" {
  type           = "ingress"
  from_port      = 0
  to_port        = 65535
  protocol        = "tcp"
  cidr_blocks     = ["0.0.0.0/0"]
  prefix_list_ids = ["pl-12c4e678"]

  security_group_id = "sg-123456"
}
```

## » Argument Reference

The following arguments are supported:

- **type** - (Required) The type of rule being created. Valid options are **ingress** (inbound) or **egress** (outbound).
- **cidr\_blocks** - (Optional) List of CIDR blocks. Cannot be specified with **source\_security\_group\_id**.
- **ipv6\_cidr\_blocks** - (Optional) List of IPv6 CIDR blocks.
- **prefix\_list\_ids** - (Optional) List of prefix list IDs (for allowing access to VPC endpoints). Only valid with **egress**.
- **from\_port** - (Required) The start port (or ICMP type number if protocol is "icmp").

- **protocol** - (Required) The protocol. If not icmp, tcp, udp, or all use the protocol number
- **security\_group\_id** - (Required) The security group to apply this rule to.
- **source\_security\_group\_id** - (Optional) The security group id to allow access to/from, depending on the **type**. Cannot be specified with **cidr\_blocks**.
- **self** - (Optional) If true, the security group itself will be added as a source to this ingress rule.
- **to\_port** - (Required) The end port (or ICMP code if protocol is "icmp").
- **description** - (Optional) Description of the rule.

## » Usage with prefix list IDs

Prefix list IDs are managed by AWS internally. Prefix list IDs are associated with a prefix list name, or service name, that is linked to a specific region. Prefix list IDs are exported on VPC Endpoints, so you can use this format:

```
resource "aws_security_group_rule" "allow_all" {
  type           = "egress"
  to_port        = 0
  protocol       = "-1"
  prefix_list_ids = ["${aws_vpc_endpoint.my_endpoint.prefix_list_id}"]
  from_port      = 0
  security_group_id = "sg-123456"
}

# ...
resource "aws_vpc_endpoint" "my_endpoint" {
  # ...
}
```

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the security group rule
- **type** - The type of rule, **ingress** or **egress**
- **from\_port** - The start port (or ICMP type number if protocol is "icmp")
- **to\_port** - The end port (or ICMP code if protocol is "icmp")
- **protocol** - The protocol used
- **description** - Description of the rule

## » **aws\_\_subnet**

Provides an VPC subnet resource.

### » **Example Usage**

```
resource "aws_subnet" "main" {
  vpc_id      = "${aws_vpc.main.id}"
  cidr_block  = "10.0.1.0/24"

  tags {
    Name = "Main"
  }
}
```

### » **Argument Reference**

The following arguments are supported:

- **availability\_zone**- (Optional) The AZ for the subnet.
- **cidr\_block** - (Required) The CIDR block for the subnet.
- **ipv6\_cidr\_block** - (Optional) The IPv6 network range for the subnet, in CIDR notation. The subnet size must use a /64 prefix length.
- **map\_public\_ip\_on\_launch** - (Optional) Specify true to indicate that instances launched into the subnet should be assigned a public IP address. Default is **false**.
- **assign\_ipv6\_address\_on\_creation** - (Optional) Specify true to indicate that network interfaces created in the specified subnet should be assigned an IPv6 address. Default is **false**
- **vpc\_id** - (Required) The VPC ID.
- **tags** - (Optional) A mapping of tags to assign to the resource.

### » **Attributes Reference**

The following attributes are exported:

- **id** - The ID of the subnet
- **availability\_zone**- The AZ for the subnet.
- **cidr\_block** - The CIDR block for the subnet.
- **vpc\_id** - The VPC ID.
- **ipv6\_association\_id** - The association ID for the IPv6 CIDR block.
- **ipv6\_cidr\_block** - The IPv6 CIDR block.

## » Import

Subnets can be imported using the `subnet_id`, e.g.

```
$ terraform import aws_subnet.public_subnet subnet-9d4a7b6c
```

## » `aws_vpc`

Provides an VPC resource.

## » Example Usage

Basic usage:

```
resource "aws_vpc" "main" {
  cidr_block = "10.0.0.0/16"
}
```

Basic usage with tags:

```
resource "aws_vpc" "main" {
  cidr_block      = "10.0.0.0/16"
  instance_tenancy = "dedicated"

  tags {
    Name = "main"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `cidr_block` - (Required) The CIDR block for the VPC.
- `instance_tenancy` - (Optional) A tenancy option for instances launched into the VPC
- `enable_dns_support` - (Optional) A boolean flag to enable/disable DNS support in the VPC. Defaults true.
- `enable_dns_hostnames` - (Optional) A boolean flag to enable/disable DNS hostnames in the VPC. Defaults false.
- `enable_classiclink` - (Optional) A boolean flag to enable/disable ClassicLink for the VPC. Only valid in regions and accounts that support EC2 Classic. See the ClassicLink documentation for more information. Defaults false.

- `enable_classiclink_dns_support` - (Optional) A boolean flag to enable/disable ClassicLink DNS Support for the VPC. Only valid in regions and accounts that support EC2 Classic.
- `assign_generated_ipv6_cidr_block` - (Optional) Requests an Amazon-provided IPv6 CIDR block with a /56 prefix length for the VPC. You cannot specify the range of IP addresses, or the size of the CIDR block. Default is `false`.
- `tags` - (Optional) A mapping of tags to assign to the resource.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the VPC
- `cidr_block` - The CIDR block of the VPC
- `instance_tenancy` - Tenancy of instances spin up within VPC.
- `enable_dns_support` - Whether or not the VPC has DNS support
- `enable_dns_hostnames` - Whether or not the VPC has DNS hostname support
- `enable_classiclink` - Whether or not the VPC has Classiclink enabled
- `main_route_table_id` - The ID of the main route table associated with this VPC. Note that you can change a VPC's main route table by using an `aws_main_route_table_association`.
- `default_network_acl_id` - The ID of the network ACL created by default on VPC creation
- `default_security_group_id` - The ID of the security group created by default on VPC creation
- `default_route_table_id` - The ID of the route table created by default on VPC creation
- `ipv6_association_id` - The association ID for the IPv6 CIDR block.
- `ipv6_cidr_block` - The IPv6 CIDR block.

## » Import

VPCs can be imported using the `vpc_id`, e.g.

```
$ terraform import aws_vpc.test_vpc vpc-a01106c2
```

## » `aws_vpc__dhcp__options`

Provides a VPC DHCP Options resource.

## » Example Usage

Basic usage:

```
resource "aws_vpc_dhcp_options" "dns_resolver" {
  domain_name_servers = ["8.8.8.8", "8.8.4.4"]
}
```

Full usage:

```
resource "aws_vpc_dhcp_options" "foo" {
  domain_name           = "service.consul"
  domain_name_servers   = ["127.0.0.1", "10.0.0.2"]
  ntp_servers           = ["127.0.0.1"]
  netbios_name_servers  = ["127.0.0.1"]
  netbios_node_type     = 2

  tags {
    Name = "foo-name"
  }
}
```

## » Argument Reference

The following arguments are supported:

- **domain\_name** - (Optional) the suffix domain name to use by default when resolving non Fully Qualified Domain Names. In other words, this is what ends up being the `search` value in the `/etc/resolv.conf` file.
- **domain\_name\_servers** - (Optional) List of name servers to configure in `/etc/resolv.conf`. If you want to use the default AWS nameservers you should set this to `AmazonProvidedDNS`.
- **ntp\_servers** - (Optional) List of NTP servers to configure.
- **netbios\_name\_servers** - (Optional) List of NETBIOS name servers.
- **netbios\_node\_type** - (Optional) The NetBIOS node type (1, 2, 4, or 8). AWS recommends to specify 2 since broadcast and multicast are not supported in their network. For more information about these node types, see RFC 2132.
- **tags** - (Optional) A mapping of tags to assign to the resource.

## » Remarks

- Notice that all arguments are optional but you have to specify at least one argument.
- `domain_name_servers`, `netbios_name_servers`, `ntp_servers` are limited by AWS to maximum four servers only.



- To actually use the DHCP Options Set you need to associate it to a VPC using `aws_vpc_dhcp_options_association`.
- If you delete a DHCP Options Set, all VPCs using it will be associated to AWS's `default` DHCP Option Set.
- In most cases unless you're configuring your own DNS you'll want to set `domain_name_servers` to `AmazonProvidedDNS`.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the DHCP Options Set.

You can find more technical documentation about DHCP Options Set in the official AWS User Guide.

## » Import

VPC DHCP Options can be imported using the `dhcp_options id`, e.g.

```
$ terraform import aws_vpc_dhcp_options.my_options dopt-d9070ebb
```

## » `aws__vpc__dhcp__options__association`

Provides a VPC DHCP Options Association resource.

## » Example Usage

```
resource "aws_vpc_dhcp_options_association" "dns_resolver" {
  vpc_id      = "${aws_vpc.foo.id}"
  dhcp_options_id = "${aws_vpc_dhcp_options.foo.id}"
}
```

## » Argument Reference

The following arguments are supported:

- `vpc_id` - (Required) The ID of the VPC to which we would like to associate a DHCP Options Set.
- `dhcp_options_id` - (Required) The ID of the DHCP Options Set to associate to the VPC.

## » Remarks

- You can only associate one DHCP Options Set to a given VPC ID.
- Removing the DHCP Options Association automatically sets AWS's default DHCP Options Set to the VPC.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the DHCP Options Set Association.

## » `aws_vpc_endpoint`

Provides a VPC Endpoint resource.

**NOTE on VPC Endpoints and VPC Endpoint Associations:** Terraform provides both standalone VPC Endpoint Associations for Route Tables - (an association between a VPC endpoint and a single `route_table_id`) and Subnets - (an association between a VPC endpoint and a single `subnet_id`) and a VPC Endpoint resource with `route_table_ids` and `subnet_ids` attributes. Do not use the same resource ID in both a VPC Endpoint resource and a VPC Endpoint Association resource. Doing so will cause a conflict of associations and will overwrite the association.

## » Example Usage

Basic usage:

```
resource "aws_vpc_endpoint" "s3" {
  vpc_id      = "${aws_vpc.main.id}"
  service_name = "com.amazonaws.us-west-2.s3"
}
```

Interface type usage:

```
resource "aws_vpc_endpoint" "ec2" {
  vpc_id            = "${aws_vpc.main.id}"
  service_name      = "com.amazonaws.us-west-2.ec2"
  vpc_endpoint_type = "Interface"

  security_group_ids = [
    "${aws_security_group.sg1.id}"
  ]
}
```

```

    private_dns_enabled = true
}

```

Custom Service Usage:

```

resource "aws_vpc_endpoint" "ptfe_service" {
    vpc_id          = "${var.vpc_id}"
    service_name     = "${var.ptfe_service}"
    vpc_endpoint_type = "Interface"

    security_group_ids = [
        "${aws_security_group.ptfe_service.id}",
    ]

    subnet_ids       = ["${local.subnet_ids}"]
    private_dns_enabled = false
}

data "aws_route53_zone" "internal" {
    name         = "vpc.internal."
    private_zone = true
    vpc_id       = "${var.vpc_id}"
}

resource "aws_route53_record" "ptfe_service" {
    zone_id = "${data.aws_route53_zone.internal.zone_id}"
    name     = "ptfe.${data.aws_route53_zone.internal.name}"
    type     = "CNAME"
    ttl      = "300"
    records  = ["${lookup(aws_vpc_endpoint.ptfe_service.dns_entry[0], "dns_name")}"]
}

```

**NOTE The dns\_entry output is a list of maps:** Terraform interpolation support for lists of maps requires the lookup and [] until full support of lists of maps is available

## » Argument Reference

The following arguments are supported:

- **vpc\_id** - (Required) The ID of the VPC in which the endpoint will be used.
- **vpc\_endpoint\_type** - (Optional) The VPC endpoint type, **Gateway** or **Interface**. Defaults to **Gateway**.
- **service\_name** - (Required) The service name, in the form **com.amazonaws.region.service** for AWS services.

- **auto\_accept** - (Optional) Accept the VPC endpoint (the VPC endpoint and service need to be in the same AWS account).
- **policy** - (Optional) A policy to attach to the endpoint that controls access to the service. Applicable for endpoints of type **Gateway**. Defaults to full access.
- **route\_table\_ids** - (Optional) One or more route table IDs. Applicable for endpoints of type **Gateway**.
- **subnet\_ids** - (Optional) The ID of one or more subnets in which to create a network interface for the endpoint. Applicable for endpoints of type **Interface**.
- **security\_group\_ids** - (Optional) The ID of one or more security groups to associate with the network interface. Required for endpoints of type **Interface**.
- **private\_dns\_enabled** - (Optional) Whether or not to associate a private hosted zone with the specified VPC. Applicable for endpoints of type **Interface**. Defaults to **false**.

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the VPC endpoint.
- **state** - The state of the VPC endpoint.
- **prefix\_list\_id** - The prefix list ID of the exposed AWS service. Applicable for endpoints of type **Gateway**.
- **cidr\_blocks** - The list of CIDR blocks for the exposed AWS service. Applicable for endpoints of type **Gateway**.
- **network\_interface\_ids** - One or more network interfaces for the VPC Endpoint. Applicable for endpoints of type **Interface**.
- **dns\_entry** - The DNS entries for the VPC Endpoint. Applicable for endpoints of type **Interface**. DNS blocks are documented below.

DNS blocks (for **dns\_entry**) support the following attributes:

- **dns\_name** - The DNS name.
- **hosted\_zone\_id** - The ID of the private hosted zone.

## » Import

VPC Endpoints can be imported using the `vpc_endpoint_id`, e.g.

```
$ terraform import aws_vpc_endpoint.endpoint1 vpce-3ecf2a57
```

## » `aws_vpc_endpoint_connection_notification`

Provides a VPC Endpoint connection notification resource. Connection notifications notify subscribers of VPC Endpoint events.

### » Example Usage

```
resource "aws_sns_topic" "topic" {
  name = "vpce-notification-topic"

  policy = <<POLICY
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "Service": "vpce.amazonaws.com"
    },
    "Action": "SNS:Publish",
    "Resource": "arn:aws:sns:*:*:vpce-notification-topic"
  }]
}
POLICY
}

resource "aws_vpc_endpoint_service" "foo" {
  acceptance_required = false
  network_load_balancer_arns = ["${aws_lb.test.arn}"]
}

resource "aws_vpc_endpoint_connection_notification" "foo" {
  vpc_endpoint_service_id = "${aws_vpc_endpoint_service.foo.id}"
  connection_notification_arn = "${aws_sns_topic.topic.arn}"
  connection_events = ["Accept", "Reject"]
}
```

### » Argument Reference

The following arguments are supported:

- `vpc_endpoint_service_id` - (Optional) The ID of the VPC Endpoint Service to receive notifications for.
- `vpc_endpoint_id` - (Optional) The ID of the VPC Endpoint to receive notifications for.

- `connection_notification_arn` - (Required) The ARN of the SNS topic for the notifications.
- `connection_events` - (Required) One or more endpoint events for which to receive notifications.

**NOTE:** One of `vpc_endpoint_service_id` or `vpc_endpoint_id` must be specified.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the VPC connection notification.
- `state` - The state of the notification.
- `notification_type` - The type of notification.

## » Import

VPC Endpoint connection notifications can be imported using the VPC endpoint connection notification id, e.g.

```
$ terraform import aws_vpc_endpoint_connection_notification.foo vpce-nfn-09e6ed3b4efba2263
```

## » `aws_vpc_endpoint_route_table_association`

Provides a resource to create an association between a VPC endpoint and routing table.

**NOTE on VPC Endpoints and VPC Endpoint Route Table Associations:** Terraform provides both a standalone VPC Endpoint Route Table Association (an association between a VPC endpoint and a single `route_table_id`) and a VPC Endpoint resource with a `route_table_ids` attribute. Do not use the same route table ID in both a VPC Endpoint resource and a VPC Endpoint Route Table Association resource. Doing so will cause a conflict of associations and will overwrite the association.

## » Example Usage

Basic usage:

```
resource "aws_vpc_endpoint_route_table_association" "private_s3" {
  vpc_endpoint_id = "${aws_vpc_endpoint.s3.id}"
  route_table_id  = "${aws_route_table.private.id}"
}
```

## » Argument Reference

The following arguments are supported:

- `vpc_endpoint_id` - (Required) The ID of the VPC endpoint with which the routing table will be associated.
- `route_table_id` - (Required) The ID of the routing table to be associated with the VPC endpoint.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the association.

## » `aws_vpc_endpoint_service`

Provides a VPC Endpoint Service resource. Service consumers can create an *Interface* VPC Endpoint to connect to the service.

**NOTE on VPC Endpoint Services and VPC Endpoint Service Allowed Principals:** Terraform provides both a standalone VPC Endpoint Service Allowed Principal resource and a VPC Endpoint Service resource with an `allowed_principals` attribute. Do not use the same principal ARN in both a VPC Endpoint Service resource and a VPC Endpoint Service Allowed Principal resource. Doing so will cause a conflict and will overwrite the association.

## » Example Usage

Basic usage:

```
resource "aws_vpc_endpoint_service" "foo" {
  acceptance_required = false
  network_load_balancer_arns = ["${aws_lb.test.arn}"]
}
```

## » Argument Reference

The following arguments are supported:

- `acceptance_required` - (Required) Whether or not VPC endpoint connection requests to the service must be accepted by the service owner - `true` or `false`.

- `network_load_balancer_arns` - (Required) The ARNs of one or more Network Load Balancers for the endpoint service.
- `allowed_principals` - (Optional) The ARNs of one or more principals allowed to discover the endpoint service.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the VPC endpoint service.
- `state` - The state of the VPC endpoint service.
- `service_name` - The service name.
- `service_type` - The service type, `Gateway` or `Interface`.
- `availability_zones` - The Availability Zones in which the service is available.
- `private_dns_name` - The private DNS name for the service.
- `base_endpoint_dns_names` - The DNS names for the service.

## » Import

VPC Endpoint Services can be imported using the `VPC_endpoint_service_id`, e.g.

```
$ terraform import aws_vpc_endpoint_service.foo vpce-svc-0f97a19d3fa8220bc
```

## » `aws_vpc_endpoint_service_allowed_principal`

Provides a resource to allow a principal to discover a VPC endpoint service.

**NOTE on VPC Endpoint Services and VPC Endpoint Service Allowed Principals:** Terraform provides both a standalone VPC Endpoint Service Allowed Principal resource and a VPC Endpoint Service resource with an `allowed_principals` attribute. Do not use the same principal ARN in both a VPC Endpoint Service resource and a VPC Endpoint Service Allowed Principal resource. Doing so will cause a conflict and will overwrite the association.

## » Example Usage

Basic usage:

```
data "aws_caller_identity" "current" {}

resource "aws_vpc_endpoint_service_allowed_principal" "allow_me_to_foo" {
```



```

    vpc_endpoint_service_id = "${aws_vpc_endpoint_service.foo.id}"
    principal_arn = "${data.aws_caller_identity.current.arn}"
}

```

## » Argument Reference

The following arguments are supported:

- `vpc_endpoint_service_id` - (Required) The ID of the VPC endpoint service to allow permission.
- `principal_arn` - (Required) The ARN of the principal to allow permissions.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the association.

## » `aws_vpc_endpoint_subnet_association`

Provides a resource to create an association between a VPC endpoint and a subnet.

**NOTE on VPC Endpoints and VPC Endpoint Subnet Associations:** Terraform provides both a standalone VPC Endpoint Subnet Association (an association between a VPC endpoint and a single `subnet_id`) and a VPC Endpoint resource with a `subnet_ids` attribute. Do not use the same subnet ID in both a VPC Endpoint resource and a VPC Endpoint Subnet Association resource. Doing so will cause a conflict of associations and will overwrite the association.

## » Example Usage

Basic usage:

```

resource "aws_vpc_endpoint_subnet_association" "sn_ec2" {
    vpc_endpoint_id = "${aws_vpc_endpoint.ec2.id}"
    subnet_id       = "${aws_subnet.sn.id}"
}

```

## » Argument Reference

The following arguments are supported:

- `vpc_endpoint_id` - (Required) The ID of the VPC endpoint with which the subnet will be associated.
- `subnet_id` - (Required) The ID of the subnet to be associated with the VPC endpoint.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the association.

## » `aws_vpc_peering_connection`

Provides a resource to manage a VPC peering connection.

**NOTE on VPC Peering Connections and VPC Peering Connection Options:** Terraform provides both a standalone VPC Peering Connection Options and a VPC Peering Connection resource with `accepter` and `requester` attributes. Do not manage options for the same VPC peering connection in both a VPC Peering Connection resource and a VPC Peering Connection Options resource. Doing so will cause a conflict of options and will overwrite the options. Using a VPC Peering Connection Options resource decouples management of the connection options from management of the VPC Peering Connection and allows options to be set correctly in cross-account scenarios.

**Note:** For cross-account (requester's AWS account differs from the acceptor's AWS account) or inter-region VPC Peering Connections use the `aws_vpc_peering_connection` resource to manage the requester's side of the connection and use the `aws_vpc_peering_connection_accepter` resource to manage the acceptor's side of the connection.

## » Example Usage

```
resource "aws_vpc_peering_connection" "foo" {
  peer_owner_id = "${var.peer_owner_id}"
  peer_vpc_id   = "${aws_vpc.bar.id}"
  vpc_id        = "${aws_vpc.foo.id}"
}
```

Basic usage with connection options:

```

resource "aws_vpc_peering_connection" "foo" {
  peer_owner_id = "${var.peer_owner_id}"
  peer_vpc_id   = "${aws_vpc.bar.id}"
  vpc_id        = "${aws_vpc.foo.id}"

  accepter {
    allow_remote_vpc_dns_resolution = true
  }

  requester {
    allow_remote_vpc_dns_resolution = true
  }
}

```

Basic usage with tags:

```

resource "aws_vpc_peering_connection" "foo" {
  peer_owner_id = "${var.peer_owner_id}"
  peer_vpc_id   = "${aws_vpc.bar.id}"
  vpc_id        = "${aws_vpc.foo.id}"
  auto_accept   = true

  tags {
    Name = "VPC Peering between foo and bar"
  }
}

```

```

resource "aws_vpc" "foo" {
  cidr_block = "10.1.0.0/16"
}

```

```

resource "aws_vpc" "bar" {
  cidr_block = "10.2.0.0/16"
}

```

Basic usage with region:

```

resource "aws_vpc_peering_connection" "foo" {
  peer_owner_id = "${var.peer_owner_id}"
  peer_vpc_id   = "${aws_vpc.bar.id}"
  vpc_id        = "${aws_vpc.foo.id}"
  peer_region   = "us-east-1"
}

```

```

resource "aws_vpc" "foo" {
  provider   = "aws.us-west-2"
  cidr_block = "10.1.0.0/16"
}

```

```
resource "aws_vpc" "bar" {
  provider    = "aws.us-east-1"
  cidr_block = "10.2.0.0/16"
}
```

## » Argument Reference

**Note:** Modifying the VPC Peering Connection options requires peering to be active. An automatic activation can be done using the `auto_accept` attribute. Alternatively, the VPC Peering Connection has to be made active manually using other means. See notes below for more information.

The following arguments are supported:

- `peer_owner_id` - (Optional) The AWS account ID of the owner of the peer VPC. Defaults to the account ID the AWS provider is currently connected to.
- `peer_vpc_id` - (Required) The ID of the VPC with which you are creating the VPC Peering Connection.
- `vpc_id` - (Required) The ID of the requester VPC.
- `auto_accept` - (Optional) Accept the peering (both VPCs need to be in the same AWS account).
- `peer_region` - (Optional) The region of the acceptor VPC of the [VPC Peering Connection]. `auto_accept` must be `false`, and use the `aws_vpc_peering_connection_accepter` to manage the acceptor side.
- `accepter` (Optional) - An optional configuration block that allows for VPC Peering Connection options to be set for the VPC that accepts the peering connection (a maximum of one).
- `requester` (Optional) - A optional configuration block that allows for VPC Peering Connection options to be set for the VPC that requests the peering connection (a maximum of one).
- `tags` - (Optional) A mapping of tags to assign to the resource.

## » Acceptor and Requester Arguments

**Note:** When enabled, the DNS resolution feature requires that VPCs participating in the peering must have support for the DNS hostnames enabled. This can be done using the `enable_dns_hostnames` attribute in the `aws_vpc` resource. See Using DNS with Your VPC user guide for more information.

- `allow_remote_vpc_dns_resolution` - (Optional) Allow a local VPC to resolve public DNS hostnames to private IP addresses when queried from instances in the peer VPC.
- `allow_classic_link_to_remote_vpc` - (Optional) Allow a local linked EC2-Classic instance to communicate with instances in a peer VPC. This

enables an outbound communication from the local ClassicLink connection to the remote VPC.

- **allow\_vpc\_to\_remote\_classic\_link** - (Optional) Allow a local VPC to communicate with a linked EC2-Classic instance in a peer VPC. This enables an outbound communication from the local VPC to the remote ClassicLink connection.

## » Timeouts

`aws_vpc_peering_connection` provides the following Timeouts configuration options:

- **create** - (Default 1 minute) Used for creating a peering connection
- **update** - (Default 1 minute) Used for peering connection modifications
- **delete** - (Default 1 minute) Used for destroying peering connections

## » Attributes Reference

The following attributes are exported:

- **id** - The ID of the VPC Peering Connection.
- **accept\_status** - The status of the VPC Peering Connection request.

## » Notes

If both VPCs are not in the same AWS account do not enable the `auto_accept` attribute. The acceptor can manage its side of the connection using the `aws_vpc_peering_connection_accepter` resource or accept the connection manually using the AWS Management Console, AWS CLI, through SDKs, etc.

## » Import

VPC Peering resources can be imported using the `vpc peering id`, e.g.

```
$ terraform import aws_vpc_peering_connection.test_connection pcx-111aaa111
```

## » `aws_vpc_peering_connection_accepter`

Provides a resource to manage the acceptor's side of a VPC Peering Connection.

When a cross-account (requester's AWS account differs from the acceptor's AWS account) or an inter-region VPC Peering Connection is created, a VPC Peering Connection resource is automatically created in the acceptor's

account. The requester can use the `aws_vpc_peering_connection` resource to manage its side of the connection and the acceptor can use the `aws_vpc_peering_connection_accepter` resource to "adopt" its side of the connection into management.

## » Example Usage

```
provider "aws" {
  region = "us-east-1"

  # Requester's credentials.
}

provider "aws" {
  alias = "peer"
  region = "us-west-2"

  # Acceptor's credentials.
}

resource "aws_vpc" "main" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_vpc" "peer" {
  provider = "aws.peer"
  cidr_block = "10.1.0.0/16"
}

data "aws_caller_identity" "peer" {
  provider = "aws.peer"
}

# Requester's side of the connection.
resource "aws_vpc_peering_connection" "peer" {
  vpc_id          = "${aws_vpc.main.id}"
  peer_vpc_id     = "${aws_vpc.peer.id}"
  peer_owner_id   = "${data.aws_caller_identity.peer.account_id}"
  peer_region     = "us-west-2"
  auto_accept     = false

  tags {
    Side = "Requester"
  }
}
```

```
# Acceptor's side of the connection.
resource "aws_vpc_peering_connection_accepter" "peer" {
  provider                = "aws.peer"
  vpc_peering_connection_id = "${aws_vpc_peering_connection.peer.id}"
  auto_accept             = true

  tags {
    Side = "Acceptor"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `vpc_peering_connection_id` - (Required) The VPC Peering Connection ID to manage.
- `auto_accept` - (Optional) Whether or not to accept the peering request. Defaults to `false`.
- `tags` - (Optional) A mapping of tags to assign to the resource.

## » Removing `aws_vpc_peering_connection_accepter` from your configuration

AWS allows a cross-account VPC Peering Connection to be deleted from either the requester's or acceptor's side. However, Terraform only allows the VPC Peering Connection to be deleted from the requester's side by removing the corresponding `aws_vpc_peering_connection` resource from your configuration. Removing a `aws_vpc_peering_connection_accepter` resource from your configuration will remove it from your statefile and management, **but will not destroy the VPC Peering Connection**.

## » Attributes Reference

All of the argument attributes except `auto_accept` are also exported as result attributes.

- `id` - The ID of the VPC Peering Connection.
- `accept_status` - The status of the VPC Peering Connection request.
- `vpc_id` - The ID of the acceptor VPC.
- `peer_vpc_id` - The ID of the requester VPC.
- `peer_owner_id` - The AWS account ID of the owner of the requester VPC.
- `peer_region` - The region of the acceptor VPC.

- **accepter** - A configuration block that describes VPC Peering Connection options set for the accepter VPC.
- **requester** - A configuration block that describes VPC Peering Connection options set for the requester VPC.

#### » **Accepter and Requester Attributes Reference**

- **allow\_remote\_vpc\_dns\_resolution** - Indicates whether a local VPC can resolve public DNS hostnames to private IP addresses when queried from instances in a peer VPC.
- **allow\_classic\_link\_to\_remote\_vpc** - Indicates whether a local ClassicLink connection can communicate with the peer VPC over the VPC Peering Connection.
- **allow\_vpc\_to\_remote\_classic\_link** - Indicates whether a local VPC can communicate with a ClassicLink connection in the peer VPC over the VPC Peering Connection.

#### » **Attributes Reference**

The following attributes are exported:

- **id** - The ID of the VPC Peering Connection.

### » **aws\_vpc\_peering\_connection\_options**

Provides a resource to manage VPC peering connection options.

**NOTE on VPC Peering Connections and VPC Peering Connection Options:** Terraform provides both a standalone VPC Peering Connection Options and a VPC Peering Connection resource with **accepter** and **requester** attributes. Do not manage options for the same VPC peering connection in both a VPC Peering Connection resource and a VPC Peering Connection Options resource. Doing so will cause a conflict of options and will overwrite the options. Using a VPC Peering Connection Options resource decouples management of the connection options from management of the VPC Peering Connection and allows options to be set correctly in cross-account scenarios.

Basic usage:

```
resource "aws_vpc" "foo" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_vpc" "bar" {
  cidr_block = "10.1.0.0/16"
```



```

}

resource "aws_vpc_peering_connection" "foo" {
  vpc_id          = "${aws_vpc.foo.id}"
  peer_vpc_id     = "${aws_vpc.bar.id}"
  auto_accept     = true
}

resource "aws_vpc_peering_connection_options" "foo" {
  vpc_peering_connection_id = "${aws_vpc_peering_connection.foo.id}"

  accepter {
    allow_remote_vpc_dns_resolution = true
  }

  requester {
    allow_vpc_to_remote_classic_link = true
    allow_classic_link_to_remote_vpc = true
  }
}

Basic cross-account usage:

provider "aws" {
  alias = "requester"

  # Requester's credentials.
}

provider "aws" {
  alias = "accepter"

  # Acceptor's credentials.
}

resource "aws_vpc" "main" {
  provider = "aws.requester"

  cidr_block = "10.0.0.0/16"

  enable_dns_support   = true
  enable_dns_hostnames = true
}

resource "aws_vpc" "peer" {
  provider = "aws.accepter"

```

```

    cidr_block = "10.1.0.0/16"

    enable_dns_support    = true
    enable_dns_hostnames = true
}

data "aws_caller_identity" "peer" {
  provider = "aws.accepter"
}

# Requester's side of the connection.
resource "aws_vpc_peering_connection" "peer" {
  provider = "aws.requester"

  vpc_id           = "${aws_vpc.main.id}"
  peer_vpc_id      = "${aws_vpc.peer.id}"
  peer_owner_id    = "${data.aws_caller_identity.peer.account_id}"
  auto_accept      = false

  tags {
    Side = "Requester"
  }
}

# Acceptor's side of the connection.
resource "aws_vpc_peering_connection_accepter" "peer" {
  provider = "aws.accepter"

  vpc_peering_connection_id = "${aws_vpc_peering_connection.peer.id}"
  auto_accept               = true

  tags {
    Side = "Acceptor"
  }
}

resource "aws_vpc_peering_connection_options" "requester" {
  provider = "aws.requester"

  # As options can't be set until the connection has been accepted
  # create an explicit dependency on the acceptor.
  vpc_peering_connection_id = "${aws_vpc_peering_connection_accepter.peer.id}"

  requester {
    allow_remote_vpc_dns_resolution = true
  }
}

```

```

}

resource "aws_vpc_peering_connection_options" "accepter" {
  provider = "aws.accepter"

  vpc_peering_connection_id = "${aws_vpc_peering_connection_accepter.peer.id}"

  accepter {
    allow_remote_vpc_dns_resolution = true
  }
}

```

## » Argument Reference

The following arguments are supported:

- **vpc\_peering\_connection\_id** - (Required) The ID of the requester VPC.
- **accepter** (Optional) - An optional configuration block that allows for VPC Peering Connection options to be set for the VPC that accepts the peering connection (a maximum of one).
- **requester** (Optional) - A optional configuration block that allows for VPC Peering Connection options to be set for the VPC that requests the peering connection (a maximum of one).

## » Acceptor and Requester Arguments

**Note:** When enabled, the DNS resolution feature requires that VPCs participating in the peering must have support for the DNS hostnames enabled. This can be done using the **enable\_dns\_hostnames** attribute in the **aws\_vpc** resource. See Using DNS with Your VPC user guide for more information.

- **allow\_remote\_vpc\_dns\_resolution** - (Optional) Allow a local VPC to resolve public DNS hostnames to private IP addresses when queried from instances in the peer VPC.
- **allow\_classic\_link\_to\_remote\_vpc** - (Optional) Allow a local linked EC2-Classic instance to communicate with instances in a peer VPC. This enables an outbound communication from the local ClassicLink connection to the remote VPC.
- **allow\_vpc\_to\_remote\_classic\_link** - (Optional) Allow a local VPC to communicate with a linked EC2-Classic instance in a peer VPC. This enables an outbound communication from the local VPC to the remote ClassicLink connection.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the VPC Peering Connection Options.

## » Import

VPC Peering Connection Options can be imported using the `vpc_peering_id`, e.g.

```
$ terraform import aws_vpc_peering_connection_options.foo pcx-111aaa111
```

## » `aws_vpn_connection`

Provides a VPN connection connected to a VPC. These objects can be connected to customer gateways, and allow you to establish tunnels between your network and the VPC.

**Note:** All arguments including `tunnel1_preshared_key` and `tunnel2_preshared_key` will be stored in the raw state as plain-text. Read more about sensitive data in state.

**Note:** The CIDR blocks in the arguments `tunnel1_inside_cidr` and `tunnel2_inside_cidr` must have a prefix of /30 and be a part of a specific range. Read more about this in the AWS documentation.

## » Example Usage

```
resource "aws_vpc" "vpc" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_vpn_gateway" "vpn_gateway" {
  vpc_id = "${aws_vpc.vpc.id}"
}

resource "aws_customer_gateway" "customer_gateway" {
  bgp_asn      = 65000
  ip_address   = "172.0.0.1"
  type         = "ipsec.1"
}

resource "aws_vpn_connection" "main" {
```

```

vpn_gateway_id      = "${aws_vpn_gateway.vpn_gateway.id}"
customer_gateway_id = "${aws_customer_gateway.customer_gateway.id}"
type                = "ipsec.1"
static_routes_only  = true
}

```

## » Argument Reference

The following arguments are supported:

- **customer\_gateway\_id** - (Required) The ID of the customer gateway.
- **static\_routes\_only** - (Optional, Default **false**) Whether the VPN connection uses static routes exclusively. Static routes must be used for devices that don't support BGP.
- **tags** - (Optional) Tags to apply to the connection.
- **type** - (Required) The type of VPN connection. The only type AWS supports at this time is "ipsec.1".
- **vpn\_gateway\_id** - (Required) The ID of the virtual private gateway.
- **tunnel1\_inside\_cidr** - (Optional) The CIDR block of the inside IP addresses for the first VPN tunnel.
- **tunnel2\_inside\_cidr** - (Optional) The CIDR block of the second IP addresses for the first VPN tunnel.
- **tunnel1\_preshared\_key** - (Optional) The preshared key of the first VPN tunnel.
- **tunnel2\_preshared\_key** - (Optional) The preshared key of the second VPN tunnel.

## » Attribute Reference

The following attributes are exported:

- **id** - The amazon-assigned ID of the VPN connection.
- **customer\_gateway\_configuration** - The configuration information for the VPN connection's customer gateway (in the native XML format).
- **customer\_gateway\_id** - The ID of the customer gateway to which the connection is attached.
- **static\_routes\_only** - Whether the VPN connection uses static routes exclusively.
- **tags** - Tags applied to the connection.
- **tunnel1\_address** - The public IP address of the first VPN tunnel.
- **tunnel1\_cgw\_inside\_address** - The RFC 6890 link-local address of the first VPN tunnel (Customer Gateway Side).
- **tunnel1\_vgw\_inside\_address** - The RFC 6890 link-local address of the first VPN tunnel (VPN Gateway Side).
- **tunnel1\_preshared\_key** - The preshared key of the first VPN tunnel.

- `tunnel1_bgp_asn` - The bgp asn number of the first VPN tunnel.
- `tunnel1_bgp_holdtime` - The bgp holdtime of the first VPN tunnel.
- `tunnel2_address` - The public IP address of the second VPN tunnel.
- `tunnel2_cgw_inside_address` - The RFC 6890 link-local address of the second VPN tunnel (Customer Gateway Side).
- `tunnel2_vgw_inside_address` - The RFC 6890 link-local address of the second VPN tunnel (VPN Gateway Side).
- `tunnel2_preshared_key` - The preshared key of the second VPN tunnel.
- `tunnel2_bgp_asn` - The bgp asn number of the second VPN tunnel.
- `tunnel2_bgp_holdtime` - The bgp holdtime of the second VPN tunnel.
- `type` - The type of VPN connection.
- `vpn_gateway_id` - The ID of the virtual private gateway to which the connection is attached.

## » Import

VPN Connections can be imported using the `vpn_connection_id`, e.g.

```
$ terraform import aws_vpn_connection.testvpnconnection vpn-40f41529
```

## » `aws_vpn_connection_route`

Provides a static route between a VPN connection and a customer gateway.

## » Example Usage

```
resource "aws_vpc" "vpc" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_vpn_gateway" "vpn_gateway" {
  vpc_id = "${aws_vpc.vpc.id}"
}

resource "aws_customer_gateway" "customer_gateway" {
  bgp_asn      = 65000
  ip_address   = "172.0.0.1"
  type         = "ipsec.1"
}

resource "aws_vpn_connection" "main" {
  vpn_gateway_id      = "${aws_vpn_gateway.vpn_gateway.id}"
  customer_gateway_id = "${aws_customer_gateway.customer_gateway.id}"
}
```

```

    type                = "ipsec.1"
    static_routes_only  = true
  }

  resource "aws_vpn_connection_route" "office" {
    destination_cidr_block = "192.168.10.0/24"
    vpn_connection_id      = "${aws_vpn_connection.main.id}"
  }

```

## » Argument Reference

The following arguments are supported:

- `destination_cidr_block` - (Required) The CIDR block associated with the local subnet of the customer network.
- `vpn_connection_id` - (Required) The ID of the VPN connection.

## » Attribute Reference

The following attributes are exported:

- `destination_cidr_block` - The CIDR block associated with the local subnet of the customer network.
- `vpn_connection_id` - The ID of the VPN connection.

## » `aws_vpn_gateway`

Provides a resource to create a VPC VPN Gateway.

## » Example Usage

```

resource "aws_vpn_gateway" "vpn_gw" {
  vpc_id = "${aws_vpc.main.id}"

  tags {
    Name = "main"
  }
}

```

## » Argument Reference

The following arguments are supported:

- `vpc_id` - (Optional) The VPC ID to create in.
- `availability_zone` - (Optional) The Availability Zone for the virtual private gateway.
- `tags` - (Optional) A mapping of tags to assign to the resource.
- `amazon_side_asn` - (Optional) The Autonomous System Number (ASN) for the Amazon side of the gateway. If you don't specify an ASN, the virtual private gateway is created with the default ASN.

## » Attributes Reference

The following attributes are exported:

- `id` - The ID of the VPN Gateway.

## » Import

VPN Gateways can be imported using the `vpn_gateway id`, e.g.

```
$ terraform import aws_vpn_gateway.testvpngateway vgw-9a4cacf3
```

## » `aws_vpn_gateway_attachment`

Provides a Virtual Private Gateway attachment resource, allowing for an existing hardware VPN gateway to be attached and/or detached from a VPC.

**Note:** The `aws_vpn_gateway` resource can also automatically attach the Virtual Private Gateway it creates to an existing VPC by setting the `vpc_id` attribute accordingly.

## » Example Usage

```
resource "aws_vpc" "network" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_vpn_gateway" "vpn" {
  tags {
    Name = "example-vpn-gateway"
  }
}

resource "aws_vpn_gateway_attachment" "vpn_attachment" {
  vpn_id      = "${aws_vpn_gateway.vpn.id}"
  vpc_id      = "${aws_vpc.network.id}"
}
```



```
    vpn_gateway_id = "${aws_vpn_gateway.vpn.id}"
}
```

See Virtual Private Cloud and Virtual Private Gateway user guides for more information.

## » Argument Reference

The following arguments are supported:

- `vpc_id` - (Required) The ID of the VPC.
- `vpn_gateway_id` - (Required) The ID of the Virtual Private Gateway.

## » Attributes Reference

The following attributes are exported:

- `vpc_id` - The ID of the VPC that Virtual Private Gateway is attached to.
- `vpn_gateway_id` - The ID of the Virtual Private Gateway.

## » Import

This resource does not support importing.

## » `aws_vpn_gateway_route_propagation`

Requests automatic route propagation between a VPN gateway and a route table.

**Note:** This resource should not be used with a route table that has the `propagating_vgws` argument set. If that argument is set, any route propagation not explicitly listed in its value will be removed.

## » Example Usage

```
resource "aws_vpn_gateway_route_propagation" "example" {
    vpn_gateway_id = "${aws_vpn_gateway.example.id}"
    route_table_id = "${aws_route_table.example.id}"
}
```

## » Argument Reference

The following arguments are required:

- `vpn_gateway_id` - The id of the `aws_vpn_gateway` to propagate routes from.
- `route_table_id` - The id of the `aws_route_table` to propagate routes into.

## » Attributes Reference

This resource does not export any additional attributes.