

FlexibleEngine Provider

The FlexibleEngine provider is used to interact with the many resources supported by FlexibleEngine. The provider needs to be configured with the proper credentials before it can be used.

Use the navigation to the left to read about the available resources.

Example Usage

```
# Configure the FlexibleEngine Provider
provider "flexibleengine" {
  user_name   = "admin"
  tenant_name = "admin"
  password    = "pwd"
  auth_url    = "http://myauthurl:5000/v2.0"
  region      = "RegionOne"
}

# Create a web server
resource "flexibleengine_compute_instance_v2" "test-server" {
  # ...
}
```

Authentication

User name + Password

```
provider "flexibleengine" {
  user_name   = "${var.user_name}"
  password    = "${var.password}"
  domain_name = "${var.domain_name}"
  tenant_name = "${var.tenant_name}"
  auth_url    = "https://iam.eu-west-0.prod-cloud-ocb.orange-business.com/v3"
  region      = "eu-west-0"
}
```

AKSK

```
provider "flexibleengine" {
  access_key = "${var.access_key}"
  secret_key = "${var.secret_key}"
  domain_name = "${var.domain_name}"
  tenant_name = "${var.tenant_name}"
  auth_url = "https://iam.eu-west-0.prod-cloud-ocb.orange-business.com/v3"
  region = "eu-west-0"
}
```

Configuration Reference

The following arguments are supported:

- `access_key` - (Optional) The access key of the FlexibleEngine cloud to use. If omitted, the `OS_ACCESS_KEY` environment variable is used.
- `secret_key` - (Optional) The secret key of the FlexibleEngine cloud to use. If omitted, the `OS_SECRET_KEY` environment variable is used.
- `auth_url` - (Required) The Identity authentication URL. If omitted, the `OS_AUTH_URL` environment variable is used.
- `region` - (Optional) The region of the FlexibleEngine cloud to use. If omitted, the `OS_REGION_NAME` environment variable is used. If `OS_REGION_NAME` is not set, then no region will be used. It should be possible to omit the region in single-region FlexibleEngine environments, but this behavior may vary depending on the FlexibleEngine environment being used.
- `user_name` - (Optional) The Username to login with. If omitted, the `OS_USERNAME` environment variable is used.
- `user_id` - (Optional) The User ID to login with. If omitted, the `OS_USER_ID` environment variable is used.
- `tenant_id` - (Optional) The ID of the Tenant (Identity v2) or Project (Identity v3) to login with. If omitted, the `OS_TENANT_ID` or `OS_PROJECT_ID` environment variables are used.
- `tenant_name` - (Optional) The Name of the Tenant (Identity v2) or Project (Identity v3) to login with. If omitted, the `OS_TENANT_NAME` or `OS_PROJECT_NAME` environment variable are used.
- `password` - (Optional) The Password to login with. If omitted, the `OS_PASSWORD` environment variable is used.
- `token` - (Optional; Required if not using `user_name` and `password`) A token is an expiring, temporary means of access issued via the Keystone service. By specifying a token, you do not have to specify a username/password combination, since the token was already created by a username/password out of band of Terraform. If omitted, the `OS_AUTH_TOKEN` environment variable is used.
- `domain_id` - (Optional) The ID of the Domain to scope to (Identity v3). If omitted, the following environment variables are checked (in this order): `OS_USER_DOMAIN_ID`, `OS_PROJECT_DOMAIN_ID`, `OS_DOMAIN_ID`.
- `domain_name` - (Optional) The Name of the Domain to scope to (Identity v3). If omitted, the following environment variables are checked (in this order): `OS_USER_DOMAIN_NAME`, `OS_PROJECT_DOMAIN_NAME`, `OS_DOMAIN_NAME`, `DEFAULT_DOMAIN`.
- `insecure` - (Optional) Trust self-signed SSL certificates. If omitted, the `OS_INSECURE` environment variable is used.

- `cacert_file` - (Optional) Specify a custom CA certificate when communicating over SSL. You can specify either a path to the file or the contents of the certificate. If omitted, the `OS_CACERT` environment variable is used.
- `cert` - (Optional) Specify client certificate file for SSL client authentication. You can specify either a path to the file or the contents of the certificate. If omitted the `OS_CERT` environment variable is used.
- `key` - (Optional) Specify client private key file for SSL client authentication. You can specify either a path to the file or the contents of the key. If omitted the `OS_KEY` environment variable is used.
- `endpoint_type` - (Optional) Specify which type of endpoint to use from the service catalog. It can be set using the `OS_ENDPOINT_TYPE` environment variable. If not set, public endpoints is used.
- `swauth` - (Optional) Set to `true` to authenticate against Swauth, a Swift-native authentication system. If omitted, the `OS_SWAUTH` environment variable is used. You must also set `username` to the Swauth/Swift username such as `username:project`. Set the `password` to the Swauth/Swift key. Finally, set `auth_url` as the location of the Swift service. Note that this will only work when used with the FlexibleEngine Object Storage resources.

Additional Logging

This provider has the ability to log all HTTP requests and responses between Terraform and the FlexibleEngine cloud which is useful for troubleshooting and debugging.

To enable these logs, set the `OS_DEBUG` environment variable to `1` along with the usual `TF_LOG=DEBUG` environment variable:

```
$ OS_DEBUG=1 TF_LOG=DEBUG terraform apply
```

If you submit these logs with a bug report, please ensure any sensitive information has been scrubbed first!

Testing and Development

In order to run the Acceptance Tests for development, the following environment variables must also be set:

- `OS_REGION_NAME` - The region in which to create the server instance.
- `OS_IMAGE_ID` or `OS_IMAGE_NAME` - a UUID or name of an existing image in Glance.
- `OS_FLAVOR_ID` or `OS_FLAVOR_NAME` - an ID or name of an existing flavor.
- `OS_POOL_NAME` - The name of a Floating IP pool.
- `OS_NETWORK_ID` - The UUID of a network in your test environment.
- `OS_EXTGW_ID` - The UUID of the external gateway.

You should be able to use any FlexibleEngine environment to develop on as long as the above environment variables are set.

Most of Terraform's FlexibleEngine support is done in a standardized Packstack all-in-one environment. You can find the scripts to build this environment here (<https://github.com/jtopjian/terraform-devstack/tree/master/packstack-standard>). The included `main.tf` file will need to be modified for your specific environment. Once it's up and running, you will have access to a standard, up-to-date FlexibleEngine environment with the latest FlexibleEngine services.

If you require access to deprecated services, such as Keystone v2 and LBaaS v1, you can use the "legacy" environment here (<https://github.com/jtopjian/terraform-devstack/tree/master/packstack-legacy>).

flexibleengine_blockstorage_volume_v2

Use this data source to get the ID of an available FlexibleEngine volume.

Example Usage

```
data "flexibleengine_blockstorage_volume_v2" "volume" {  
  name = "test_volume"  
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Volume client. If omitted, the `region` argument of the provider is used.
- `name` - (Optional) The name of the volume.
- `status` - (Optional) The status of the volume.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the volume.

flexibleengine_cce_cluster_v3

Provides details about all clusters and obtains certificate for accessing cluster information.

Example Usage

```
variable "cluster_name" { }
variable "cluster_id" { }
variable "vpc_id" { }

data "flexibleengine_cce_cluster_v3" "cluster" {
  name = "${var.cluster_name}"
  id = "${var.cluster_id}"
  status = "Available"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Optional) The Name of the cluster resource.
- `id` - (Optional) The ID of container cluster.
- `status` - (Optional) The state of the cluster.
- `cluster_type` - (Optional) Type of the cluster. Possible values: VirtualMachine, BareMetal or Windows

Attributes Reference

All above argument parameters can be exported as attribute parameters along with attribute reference:

- `billingMode` - Charging mode of the cluster.
- `description` - Cluster description.
- `name` - The name of the cluster in string format.
- `id` - The ID of the cluster.
- `flavor_id` - The cluster specification in string format.
- `cluster_version` - The version of cluster in string format.
- `container_network_cidr` - The container network segment.
- `container_network_type` - The container network type: overlay_l2 , underlay_ipvlan or vpc-router.
- `subnet_id` - The ID of the subnet used to create the node.

- `highway_subnet_id` - The ID of the high speed network used to create bare metal nodes.
- `internal_endpoint` - The internal network address.
- `external_endpoint` - The external network address.
- `external_apig_endpoint` - The endpoint of the cluster to be accessed through API Gateway.
 - `security_group_id` - Security group ID of the cluster.

Data Source: flexibleengine_cce_node_ids_v3

flexibleengine_cce_node_ids_v3 provides a list of node ids for a CCE cluster. This resource can be useful for getting back a list of node ids for a CCE cluster.

Example Usage

```
data "flexibleengine_cce_node_ids_v3" "node_ids" {
  cluster_id = "${var.cluster_id}"
}
```

Argument Reference

The following arguments are supported:

- `cluster_id` (Required) - Specifies the CCE cluster ID used as the query filter.

Attributes Reference

The following attributes are exported:

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

flexibleengine_cce_node_v3

To get the specified node in a cluster.

Example Usage

```
variable "cluster_id" { }  
variable "node_id" { }  
  
data "flexibleengine_cce_node_v3" "node" {  
  cluster_id = "${var.cluster_id}"  
  node_id = "${var.node_id}"  
}
```

Argument Reference

The following arguments are supported:

- `cluster_id` - (Required) The id of container cluster.
- `name` - (Optional) - Name of the node.
- `node_id` - (Optional) - The id of the node.
- `status` - (Optional) - The state of the node.

Attributes Reference

All above argument parameters can be exported as attribute parameters along with attribute reference:

- `flavor_id` - The flavor id to be used.
- `availability_zone` - Available partitions where the node is located.
- `key_pair` - Key pair name when logging in to select the key pair mode.
- `billing_mode` - Node's billing mode: The value is 0 (on demand).
- `charge_mode` - Bandwidth billing type.
- `bandwidth_size` - Bandwidth (Mbit/s), in the range of [1, 2000].
- `extendparam` - Extended parameters.
- `eip_ids` - List of existing elastic IP IDs.
- `server_id` - The node's virtual machine ID in ECS.
- `public_ip` - Elastic IP parameters of the node.

- `private_ip` - Private IP of the node
- `ip_type` - Elastic IP address type.
- `share_type` - The bandwidth sharing type.

NOTE: This parameter is mandatory when `share_type` is set to PER and is optional when `share_type` is set to WHOLE with an ID specified.

Enumerated values: PER (indicates exclusive bandwidth) and WHOLE (indicates sharing)

root_volumes

- `disk_size` - Disk size in GB.
- `volumetype` - Disk type.

data_volumes

- `disk_size` - Disk size in GB.
- `volumetype` - Disk type.

Data Source: flexibleengine_compute_bms_flavors_v2

flexibleengine_compute_bms_flavors_v2 used to query flavors of BMSs.

Example Usage

```
variable "flavor_id" { }
variable "disk_size" { }

data "flexibleengine_compute_bms_flavors_v2" "Query_BMS_flavors"
{
  id = "${var.bms_id}",
  min_disk = "${var.disk_size}",
  sort_key = "id",
  sort_dir = "desc",
}
```

Argument Reference

The arguments of this data source act as filters for querying the BMSs details.

- `name` - (Optional) - The name of the BMS flavor.
- `id` (Optional) - The BMS flavor id.
- `min_ram` (Optional) - The minimum memory size in MB. Only the BMSs with the memory size greater than or equal to the minimum size can be queried.
- `min_disk` (Optional) - The minimum disk size in GB. Only the BMSs with a disk size greater than or equal to the minimum size can be queried.
- `sort_key` (Optional) - The sorting field. The default value is `flavorid`. The other values are `name`, `memory_mb`, `vcpus`, `root_gb`, or `flavorid`.
- `sort_dir` (Optional) - The sorting order, which can be `ascending (asc)` or `descending (desc)`. The default value is `asc`.

Attributes Reference

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

- `ram` - It is the memory size (in MB) of the flavor.
- `vcpus` - It is the number of CPU cores in the BMS flavor.
- `disk` - Specifies the disk size (GB) in the BMS flavor.
- `swap` - This is a reserved attribute.

- `rx_tx_factor` - This is a reserved attribute.

Data Source: flexibleengine_compute_bms_keypairs_v2

flexibleengine_compute_bms_keypairs_v2 used to query SSH key pairs.

Example Usage

```
variable "keypair_name" {}

data "flexibleengine_compute_bms_keypairs_v2" "Query_BMS_keypair"
{
  name = "${var.keypair_name}"
}
```

Argument Reference

The arguments of this data source act as filters for querying the BMSs details.

- `name` - (Required) - It is the key pair name.

Attributes Reference

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

- `public_key` - It gives the information about the public key in the key pair.
- `fingerprint` - It is the fingerprint information about the key pair.

Data Source: flexibleengine_compute_bms_nic_v2

flexibleengine_compute_bms_nic_v2 used to query information about a BMS NIC based on the NIC ID.

Example Usage

```
variable "bms_id" {}
variable "nic_id" {}

data "flexibleengine_compute_bms_nic_v2" "Query_BMS_Nic"
{
  server_id = "${var.bms_id}",
  id = "${var.nic_id}",
}
```

Argument Reference

The arguments of this data source act as filters for querying the BMSs details.

- `server_id` - (Required) - This is the unique BMS id.
- `id` - (Optional) - The ID of the NIC.
- `status` - (Optional) - The NIC port status.

Attributes Reference

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

- `mac_address` - It is NIC's mac address.
- `fixed_ips` - The NIC IP address.
- `network_id` - The ID of the network to which the NIC port belongs.

Data Source: flexibleengine_compute_bms_server_v2

flexibleengine_compute_bms_server_v2 used to query a BMS or BMSs details.

Example Usage

```
variable "bms_id" {}
variable "bms_name" {}

data "flexibleengine_compute_bms_server_v2" "Query_BMS"
{
  id = "${var.bms_id}",
  name = "${var.bms_name}"
}
```

Argument Reference

The arguments of this data source act as filters for querying the BMSs details.

- `id` - (Optional) - The unique ID of the BMS.
- `user_id` (Optional) - The ID of the user to which the BMS belongs.
- `name` (Optional) - The name of BMS.
- `status` (Optional) - The BMS status.
- `host_status` (Optional) - The nova-compute status: **UP**, **UNKNOWN**, **DOWN**, **MAINTENANCE** and **Null**.
- `key_name` (Optional) - It is the SSH key name.
- `flavor_id` (Optional) - It gives the BMS flavor information.
- `image_id` (Optional) - The BMS image.

Attributes Reference

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

- `host_id` - It is the host ID of the BMS.
- `progress` - This is a reserved attribute.
- `metadata` - The BMS metadata is specified.
- `access_ip_v4` - This is a reserved attribute.
- `access_ip_v6` - This is a reserved attribute.

- `addresses` - It gives the BMS network address.
- `security_groups` - The list of security groups to which the BMS belongs.
- `tags` - Specifies the BMS tag.
- `locked` - It specifies whether a BMS is locked, true: The BMS is locked, false: The BMS is not locked.
- `config_drive` - This is a reserved attribute.
- `availability_zone` - Specifies the AZ ID.
- `description` - Provides supplementary information about the pool.
- `kernel_id` - The UUID of the kernel image when the AMI image is used.
- `hypervisor_hostname` - It is the name of a host on the hypervisor.
- `instance_name` - Instance name is specified.

Data Source: flexibleengine_csbs_backup_policy_v1

The FlexibleEngine CSBS Backup Policy data source allows access of backup Policy resources.

Example Usage

```
variable "policy_id" { }

data "flexibleengine_csbs_backup_policy_v1" "csbs_policy" {
  id = "${var.policy_id}"
}
```

Argument Reference

The following arguments are supported:

- `id` - (Optional) Specifies the ID of backup policy.
- `name` - (Optional) Specifies the backup policy name.
- `status` - (Optional) Specifies the backup policy status.

Attributes Reference

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

- `description` - Specifies the backup policy description.
- `provider_id` - Provides the Backup provider ID.
- `parameters` - Specifies the parameters of a backup policy.
- `scheduled_operation` block supports the following arguments:
 - `name` - Specifies Scheduling period name.
 - `description` - Specifies Scheduling period description.
 - `enabled` - Specifies whether the scheduling period is enabled.
 - `max_backups` - Specifies maximum number of backups that can be automatically created for a backup object.
 - `retention_duration_days` - Specifies duration of retaining a backup, in days.
 - `permanent` - Specifies whether backups are permanently retained.
 - `trigger_pattern` - Specifies Scheduling policy of the scheduler.
 - `operation_type` - Specifies Operation type, which can be backup.

- `id` - Specifies Scheduling period ID.
- `trigger_id` - Specifies Scheduler ID.
- `trigger_name` - Specifies Scheduler name.
- `trigger_type` - Specifies Scheduler type.
- `resource` block supports the following arguments:
 - `id` - Specifies the ID of the object to be backed up.
 - `type` - Entity object type of the backup object.
 - `name` - Specifies backup object name.

Data Source: flexibleengine_csbs_backup_v1

The FlexibleEngine CSBS Backup data source allows access of backup resources.

Example Usage

```
variable "backup_name" { }

data "flexibleengine_csbs_backup_v1" "csbs" {
  backup_name = "${var.backup_name}"
}
```

Argument Reference

The following arguments are supported:

- `id` - (Optional) Specifies the ID of backup.
- `backup_name` - (Optional) Specifies the backup name.
- `status` - (Optional) Specifies the backup status.
- `resource_name` - (Optional) Specifies the backup object name.
- `backup_record_id` - (Optional) Specifies the backup record ID.
- `resource_type` - (Optional) Specifies the type of backup objects.
- `resource_id` - (Optional) Specifies the backup object ID.
- `policy_id` - (Optional) Specifies the Policy Id.
- `vm_ip` - (Optional) Specifies the ip of VM.

Attributes Reference

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

- `description` - Provides the backup description.
- `auto_trigger` - Specifies whether automatic trigger is enabled.
- `average_speed` - Specifies average speed.
- `size` - Specifies the backup capacity.
- `volume_backups`
 - `space_saving_ratio` - Specifies the space saving rate.

- `volume_backups` block supports the following arguments:
 - `status` - Status of backup Volume.
 - `space_saving_ratio` - Specifies space saving rate.
 - `name` - It gives EVS disk backup name.
 - `bootable` - Specifies whether the disk is bootable.
 - `average_speed` - Specifies the average speed.
 - `source_volume_size` - Shows source volume size in GB.
 - `source_volume_id` - It specifies source volume ID.
 - `incremental` - Shows whether incremental backup is used.
 - `snapshot_id` - ID of snapshot.
 - `source_volume_name` - Specifies source volume name.
 - `image_type` - It specifies backup. The default value is backup.
 - `id` - Specifies Cinder backup ID.
 - `size` - Specifies accumulated size (MB) of backups.
- `vm_metadata` block supports the following arguments:
 - `name` - Name of backup data.
 - `eip` - Specifies elastic IP address of the ECS.
 - `cloud_service_type` - Specifies ECS type.
 - `ram` - Specifies memory size of the ECS, in MB.
 - `vcpus` - Specifies CPU cores corresponding to the ECS.
 - `private_ip` - It specifies internal IP address of the ECS.
 - `disk` - Shows system disk size corresponding to the ECS specifications.
 - `image_type` - Specifies image type.

Data Source: flexibleengine_cts_tracker_v1

CTS Tracker data source allows access of Cloud Tracker.

Example Usage

```
variable "bucket_name" { }

data "flexibleengine_cts_tracker_v1" "tracker_v1" {
  bucket_name = "${var.bucket_name}"
}
```

Argument Reference

The following arguments are supported:

- `tracker_name` - (Optional) The tracker name.
- `bucket_name` - (Optional) The OBS bucket name for a tracker.
- `file_prefix_name` - (Optional) The prefix of a log that needs to be stored in an OBS bucket.
- `status` - (Optional) Status of a tracker.

Attributes Reference

All above argument parameters can be exported as attribute parameters.

flexibleengine_dcs_az_v1

Use this data source to get the ID of an available Flexibleengine dcs az.

Example Usage

```
data "flexibleengine_dcs_az_v1" "az1" {  
  name = "AZ1"  
  port = "8004"  
  code = "sa-chile-1a"  
}
```

Argument Reference

- `name` - (Required) Indicates the name of an AZ.
- `code` - (Optional) Indicates the code of an AZ.
- `port` - (Required) Indicates the port number of an AZ.

Attributes Reference

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

- `name` - See Argument Reference above.
- `code` - See Argument Reference above.
- `port` - See Argument Reference above.

flexibleengine_dcs_maintainwindow_v1

Use this data source to get the ID of an available Flexibleengine dcs maintainwindow.

Example Usage

```
data "flexibleengine_dcs_maintainwindow_v1" "maintainwindow1" {  
  seq = 1  
}
```

Argument Reference

- `seq` - (Required) Indicates the sequential number of a maintenance time window.
- `begin` - (Optional) Indicates the time at which a maintenance time window starts.
- `end` - (Required) Indicates the time at which a maintenance time window ends.
- `default` - (Required) Indicates whether a maintenance time window is set to the default time segment.

Attributes Reference

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

- `begin` - See Argument Reference above.
- `end` - See Argument Reference above.
- `default` - See Argument Reference above.

flexibleengine_dcs_product_v1

Use this data source to get the ID of an available Flexibleengine dcs product.

Example Usage

```
data "flexibleengine_dcs_product_v1" "product1" {
  engine = "kafka"
  version = "1.1.0"
  instance_type = "cluster"
  partition_num = 300
  storage = 600
  storage_spec_code = "dcs.physical.storage.high"
}
```

Argument Reference

- `engine` - (Required) Indicates the name of a message engine.
- `version` - (Optional) Indicates the version of a message engine.
- `instance_type` - (Required) Indicates an instance type. Options: "single" and "cluster"
- `vm_specification` - (Optional) Indicates VM specifications.
- `storage` - (Optional) Indicates the message storage space.
- `bandwidth` - (Optional) Indicates the baseline bandwidth of a Kafka instance.
- `partition_num` - (Optional) Indicates the maximum number of topics that can be created for a Kafka instance.
- `storage_spec_code` - (Optional) Indicates an I/O specification.
- `io_type` - (Optional) Indicates an I/O type.
- `node_num` - (Optional) Indicates the number of nodes in a cluster.

Attributes Reference

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

- `engine` - See Argument Reference above.
- `version` - See Argument Reference above.
- `instance_type` - See Argument Reference above.
- `vm_specification` - See Argument Reference above.

- bandwidth - See Argument Reference above.
- partition_num - See Argument Reference above.
- storage_spec_code - See Argument Reference above.
- io_type - See Argument Reference above.
- node_num - See Argument Reference above.

flexibleengine_dds_flavors_v3

Use this data source to get the ID of an available FlexibleEngine dds flavor.

Example Usage

```
data "flexibleengine_dds_flavors_v3" "flavor" {  
  region = "eu-west-0"  
  engine_name = "DDS-Community"  
}
```

Argument Reference

- `region` - (Optional) The region in which to obtain the V3 dds client.
- `engine_name` - (Optional) The engine name of the dds, now only DDS-Community is supported.
- `speccode` - (Optional) The spec code of a dds flavor.

Available value for attributes

engine_name | type | vcpus | ram | speccode ---- | --- | --- DDS-Community | mongos | 1 | 4 |
dds.mongodb.s3.medium.4.mongos DDS-Community | mongos | 2 | 8 | dds.mongodb.s3.large.4.mongos DDS-Community
| mongos | 4 | 16 | dds.mongodb.s3.xlarge.4.mongos DDS-Community | mongos | 8 | 32 |
dds.mongodb.s3.2xlarge.4.mongos DDS-Community | mongos | 16 | 64 | dds.mongodb.s3.4xlarge.4.mongos DDS-
Community | shard | 1 | 4 | dds.mongodb.s3.medium.4.shard DDS-Community | shard | 2 | 8 |
dds.mongodb.s3.large.4.shard DDS-Community | shard | 4 | 16 | dds.mongodb.s3.xlarge.4.shard DDS-Community | shard
| 8 | 32 | dds.mongodb.s3.2xlarge.4.shard DDS-Community | shard | 16 | 64 | dds.mongodb.s3.4xlarge.4.shard DDS-
Community | config | 2 | 4 | dds.mongodb.s3.large.2.config DDS-Community | replica | 1 | 4 |
dds.mongodb.s3.medium.4.repset DDS-Community | replica | 2 | 8 | dds.mongodb.s3.large.4.repset DDS-Community |
replica | 4 | 16 | dds.mongodb.s3.xlarge.4.repset DDS-Community | replica | 8 | 32 | dds.mongodb.s3.2xlarge.4.repset
DDS-Community | replica | 16 | 64 | dds.mongodb.s3.4xlarge.4.repset

Attributes Reference

- `region` - See Argument Reference above.
- `engine_name` - See Argument Reference above.
- `speccode` - See Argument Reference above.
- `type` - The type of the dds flavor.
- `vcpus` - The vcpus of the dds flavor.

- ram - The ram of the dds flavor.

flexibleengine_dns_zone_v2

Use this data source to get the ID of an available FlexibleEngine DNS zone.

Example Usage

```
data "flexibleengine_dns_zone_v2" "zone_1" {  
  name = "example.com"  
}
```

Argument Reference

- `region` - (Optional) The region in which to obtain the V2 DNS client. A DNS client is needed to retrieve zone ids. If omitted, the `region` argument of the provider is used.
- `name` - (Optional) The name of the zone.
- `description` - (Optional) A description of the zone.
- `email` - (Optional) The email contact for the zone record.
- `status` - (Optional) The zone's status.
- `ttl` - (Optional) The time to live (TTL) of the zone.
- `zone_type` - (Optional) The type of the zone. Can either be `public` or `private`.

Attributes Reference

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

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `email` - See Argument Reference above.
- `zone_type` - See Argument Reference above.
- `ttl` - See Argument Reference above.
- `description` - See Argument Reference above.
- `status` - See Argument Reference above.
- `masters` - An array of master DNS servers.
- `serial` - The serial number of the zone.
- `pool_id` - The ID of the pool hosting the zone.

- `project_id` - The project ID that owns the zone.

flexibleengine_images_image_v2

Use this data source to get the ID of an available FlexibleEngine image.

Example Usage

```
data "flexibleengine_images_image_v2" "ubuntu" {
  name = "Ubuntu 16.04"
  most_recent = true
}
```

Argument Reference

- `region` - (Optional) The region in which to obtain the V2 Glance client. A Glance client is needed to create an Image that can be used with a compute instance. If omitted, the `region` argument of the provider is used.
- `most_recent` - (Optional) If more than one result is returned, use the most recent image.
- `name` - (Optional) The name of the image.
- `owner` - (Optional) The owner (UUID) of the image.
- `size_min` - (Optional) The minimum size (in bytes) of the image to return.
- `size_max` - (Optional) The maximum size (in bytes) of the image to return.
- `sort_direction` - (Optional) Order the results in either `asc` or `desc`.
- `sort_key` - (Optional) Sort images based on a certain key. Defaults to `name`.
- `tag` - (Optional) Search for images with a specific tag.
- `visibility` - (Optional) The visibility of the image. Must be one of "public", "private", "community", or "shared". Defaults to "private".

Attributes Reference

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

- `checksum` - The checksum of the data associated with the image.
- `created_at` - The date the image was created.
- `container_format` : The format of the image's container.
- `disk_format` : The format of the image's disk.
- `file` - the trailing path after the glance endpoint that represent the location of the image or the path to retrieve it.
- `metadata` - The metadata associated with the image. Image metadata allow for meaningfully define the image

properties and tags. See <http://docs.openstack.org/developer/glance/metadefs-concepts.html>
(<http://docs.openstack.org/developer/glance/metadefs-concepts.html>).

- `min_disk_gb` : The minimum amount of disk space required to use the image.
- `min_ram_mb` : The minimum amount of ram required to use the image.
- `protected` - Whether or not the image is protected.
- `schema` - The path to the JSON-schema that represent the image or image
- `size_bytes` - The size of the image (in bytes).
- `tags` - See Argument Reference above.
- `update_at` - The date the image was last updated.

flexibleengine_kms_data_key_v1

Use this data source to get the plaintext and the ciphertext of an available FlexibleEngine KMS DEK (data encryption key).

Example Usage

```
resource "flexibleengine_kms_key_v1" "key1" {
  key_alias      = "key_1"
  pending_days  = "7"
  key_description = "first test key"
}

data "flexibleengine_kms_data_key_v1" "kms_datakey1" {
  key_id          = "${flexibleengine_kms_key_v1.key1.id}"
  datakey_length = "512"
}
```

Argument Reference

- `key_id` - (Required) The globally unique identifier for the key. Changing this gets the new data encryption key.
- `encryption_context` - (Optional) The value of this parameter must be a series of "key:value" pairs used to record resource context information. The value of this parameter must not contain sensitive information and must be within 8192 characters in length. Example: {"Key1":"Value1","Key2":"Value2"}
- `datakey_length` - (Required) Number of bits in the length of a DEK (data encryption keys). The maximum number is 512. Changing this gets the new data encryption key.

Attributes Reference

`id` is set to the date of the found data key. In addition, the following attributes are exported:

- `plain_text` - The plaintext of a DEK is expressed in hexadecimal format, and two characters indicate one byte.
- `cipher_text` - The ciphertext of a DEK is expressed in hexadecimal format, and two characters indicate one byte.

flexibleengine_kms_key_v1

Use this data source to get the ID of an available FlexibleEngine KMS key.

Example Usage

```
data "flexibleengine_kms_key_v1" "key_1" {
  key_alias      = "test_key"
  key_description = "test key description"
  key_state      = "2"
  key_id         = "af650527-a0ff-4527-aef3-c493df1f3012"
  realm          = "cn-north-1"
  default_key_flag = "0"
  domain_id      = "b168fe00ff56492495a7d22974df2d0b"
  origin         = "kms"
}
```

Argument Reference

- `key_alias` - (Optional) The alias in which to create the key. It is required when we create a new key. Changing this gets the new key.
- `key_description` - (Optional) The description of the key as viewed in FlexibleEngine console. Changing this gets a new key.
- `realm` - (Optional) Region where a key resides. Changing this gets a new key.
- `key_id` - (Optional) The globally unique identifier for the key. Changing this gets the new key.
- `default_key_flag` - (Optional) Identification of a Master Key. The value "1" indicates a Default Master Key, and the value "0" indicates a key. Changing this gets a new key.
- `key_state` - (Optional) The state of a key. "1" indicates that the key is waiting to be activated. "2" indicates that the key is enabled. "3" indicates that the key is disabled. "4" indicates that the key is scheduled for deletion. Changing this gets a new key.
- `domain_id` - (Optional) - ID of a user domain for the key. Changing this gets a new key.
- `origin` - Origin of a key. such as: kms. Changing this gets a new key.

Attributes Reference

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

- `key_alias` - See Argument Reference above.
- `key_description` - See Argument Reference above.

- realm - See Argument Reference above.
- key_id - See Argument Reference above.
- default_key_flag - See Argument Reference above.
- origin - See Argument Reference above.
- scheduled_deletion_date - Scheduled deletion time (time stamp) of a key.
- domain_id - See Argument Reference above.
- expiration_time - Expiration time.
- creation_date - Creation time (time stamp) of a key.
- key_state - See Argument Reference above.

flexibleengine_networking_network_v2

Use this data source to get the ID of an available FlexibleEngine network.

Example Usage

```
data "flexibleengine_networking_network_v2" "network" {
  name = "tf_test_network"
}
```

Argument Reference

- `region` - (Optional) The region in which to obtain the V2 Neutron client. A Neutron client is needed to retrieve networks ids. If omitted, the `region` argument of the provider is used.
- `network_id` - (Optional) The ID of the network.
- `name` - (Optional) The name of the network.
- `matching_subnet_cidr` - (Optional) The CIDR of a subnet within the network.
- `tenant_id` - (Optional) The owner of the network.

Attributes Reference

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

- `admin_state_up` - (Optional) The administrative state of the network.
- `name` - See Argument Reference above.
- `region` - See Argument Reference above.
- `shared` - (Optional) Specifies whether the network resource can be accessed by any tenant or not.

flexibleengine_networking_secgroup_v2

Use this data source to get the ID of an available FlexibleEngine security group.

Example Usage

```
data "flexibleengine_networking_secgroup_v2" "secgroup" {
  name = "tf_test_secgroup"
}
```

Argument Reference

- `region` - (Optional) The region in which to obtain the V2 Neutron client. A Neutron client is needed to retrieve security groups ids. If omitted, the `region` argument of the provider is used.
- `secgroup_id` - (Optional) The ID of the security group.
- `name` - (Optional) The name of the security group.
- `tenant_id` - (Optional) The owner of the security group.

Attributes Reference

`id` is set to the ID of the found security group. In addition, the following attributes are exported:

- `name` - See Argument Reference above.
- `description` - The description of the security group.
- `region` - See Argument Reference above.

flexibleengine_rds_flavors_v1

Use this data source to get the ID of an available FlexibleEngine rds flavor.

Example Usage

```
data "flexibleengine_rds_flavors_v1" "flavor" {  
  region = "eu-de"  
  datastore_name = "PostgreSQL"  
  datastore_version = "9.5.5"  
  speccode = "rds.pg.s1.medium"  
}
```

Argument Reference

- `region` - (Required) The region in which to obtain the V1 rds client.
- `datastore_name` - (Required) The datastore name of the rds.
- `datastore_version` - (Required) The datastore version of the rds.
- `speccode` - (Optional) The spec code of a rds flavor.

Available value for attributes

<code>datastore_name</code>	<code>datastore_version</code>	<code>speccode</code>
PostgreSQL	9.5.5	rds.pg.s1.xlarge rds.pg.m1.2xlarge rds.pg.c2.xlarge rds.pg.s1.medium
	9.6.3	rds.pg.c2.medium rds.pg.s1.large rds.pg.c2.large rds.pg.m1.large
	9.6.5	rds.pg.s1.2xlarge rds.pg.m1.xlarge
MySQL	5.6.33	rds.mysql.s1.medium rds.mysql.s1.large rds.mysql.s1.xlarge rds.mysql.s1.2xlarge
	5.6.30	rds.mysql.m1.2xlarge rds.mysql.c2.medium rds.mysql.c2.large
	5.6.34	rds.mysql.c2.xlarge rds.mysql.m1.large rds.mysql.m1.xlarge
	5.6.35	
	5.7.17	
SQLServer	2014 SP2 SE	rds.mssql.s1.xlarge rds.mssql.m1.2xlarge rds.mssql.c2.xlarge
		rds.mssql.s1.2xlarge rds.mssql.m1.xlarge

Attributes Reference

`id` is set to the ID of the found rds flavor. In addition, the following attributes are exported:

- `region` - See Argument Reference above.
- `datastore_name` - See Argument Reference above.

- `datastore_version` - See Argument Reference above.
- `speccode` - See Argument Reference above.
- `name` - The name of the rds flavor.
- `ram` - The name of the rds flavor.

flexibleengine_rds_flavors_v3

Use this data source to get available FlexibleEngine rds flavors.

Example Usage

```
data "flexibleengine_rds_flavors_v3" "flavor" {
  db_type = "PostgreSQL"
  db_version = "9.5"
  instance_mode = "ha"
}
```

Argument Reference

- `db_type` - (Required) Specifies the DB engine. Value: MySQL, PostgreSQL, SQLServer.
- `db_version` - (Required) Specifies the database version. MySQL databases support MySQL 5.6 and 5.7. PostgreSQL databases support PostgreSQL 9.5 and 9.6. Microsoft SQL Server databases support 2014_SE, 2016_SE, and 2016_EE.
- `instance_mode` - (Required) The mode of instance. Value: ha(indicates primary/standby instance), single(indicates single instance)

Attributes Reference

In addition, the following attributes are exported:

- `flavors` - Indicates the flavors information. Structure is documented below.

The `flavors` block contains:

- `name` - The name of the rds flavor.
- `vcpus` - Indicates the CPU size.
- `memory` - Indicates the memory size in GB.
- `mode` - See 'instance_mode' above.

Data Source: flexibleengine_rts_software_config_v1

The RTS Software Config data source provides details about a specific RTS Software Config.

Example Usage

```
variable "config_name" {}

variable "server_id" {}

data "flexibleengine_rts_software_config_v1" "myconfig" {
  id = "${var.config_name}"
}

resource "flexibleengine_rts_software_deployment_v1" "mydeployment" {
  config_id = "${data.flexibleengine_rts_software_config_v1.myconfig.id}"
  server_id = "${var.server_id}"
}
```

Argument Reference

The following arguments are supported:

- `id` - (Optional) The id of the software configuration.
- `name` - (Optional) The name of the software configuration.

Attributes Reference

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

- `group` - The namespace that groups this software configuration by when it is delivered to a server.
- `inputs` - A list of software configuration inputs.
- `outputs` - A list of software configuration outputs.
- `config` - The software configuration code.
- `options` - The software configuration options.

Data Source: flexibleengine_rts_stack_resource_v1

The FlexibleEngine RTS Stack Resource data source allows access to stack resource metadata.

Example Usage

```
variable "stack_name" { }
variable "resource_name" { }

data "flexibleengine_rts_stack_resource_v1" "stackresource" {
  stack_name = "${var.stack_name}"
  resource_name = "${var.resource_name}"
}
```

Argument Reference

The following arguments are supported:

- `stack_name` - (Required) The unique stack name.
- `resource_name` - (Optional) The name of a resource in the stack.
- `physical_resource_id` - (Optional) The physical resource ID.
- `resource_type` - (Optional) The resource type.

Attributes Reference

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

- `logical_resource_id` - The logical resource ID.
- `resource_status` - The status of the resource.
- `resource_status_reason` - The resource operation reason.
- `required_by` - Specifies the resource dependency.

Data Source: flexibleengine_rts_stack_v1

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

Example Usage

```
variable "stack_name" { }

data "flexibleengine_rts_stack_v1" "mystack" {
  name = "${var.stack_name}"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the stack.

Attributes Reference

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

- `id` - A unique identifier of the stack.
- `capabilities` - List of stack capabilities for stack.
- `notification_topics` - List of notification topics for stack.
- `status` - Specifies the stack status.
- `disable_rollback` - Whether the rollback of the stack is disabled when stack creation fails.
- `outputs` - A list of stack outputs.
- `parameters` - A map of parameters that specify input parameters for the stack.
- `template_body` - Structure containing the template body.
- `timeout_mins` - Specifies the timeout duration.

flexibleengine_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

```
data "flexibleengine_s3_bucket_object" "b" {
  bucket = "my-test-bucket"
  key    = "hello-world.zip"
}
```

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
- `range` - (Optional) Obtains the specified range bytes of an object. The value is a range starting from 0 to maximum object length minus one. If the range is invalid, all object data is returned.
- `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 (https://en.wikipedia.org/wiki/HTTP_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, 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.
- `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. S3 stores the value of this header in the object metadata.

Data Source: flexibleengine_sfs_file_system_v2

Provides information about an Shared File System (SFS).

Example Usage

```
variable "share_name" { }

variable "share_id" { }

data "flexibleengine_sfs_file_system_v2" "shared_file"
{
  name = "${var.share_name}"
  id = "${var.share_id}"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Optional) The name of the shared file system.
- `id` - (Optional) The UUID of the shared file system.
- `status` - (Optional) The status of the shared file system.

Attributes Reference

The following attributes are exported:

- `availability_zone` - The availability zone name.
- `size` - The size (GB) of the shared file system.
- `share_type` - The storage service type for the shared file system, such as high-performance storage (composed of SSDs) or large-capacity storage (composed of SATA disks).
- `status` - The status of the shared file system.
- `host` - The host name of the shared file system.
- `is_public` - The level of visibility for the shared file system.
- `share_proto` - The protocol for sharing file systems.
- `volume_type` - The volume type.
- `metadata` - Metadata key and value pairs as a dictionary of strings.
- `export_location` - The path for accessing the shared file system.

- `export_locations` - The list of mount locations.
- `access_level` - The level of the access rule.
- `access_rules_status` - The status of the share access rule.
- `access_type` - The type of the share access rule.
- `access_to` - The access that the back end grants or denies.
- `state` - The status of the access rule.
- `share_access_id` - The UUID of the share access rule.
- `mount_id` - The UUID of the mount location of the shared file system.
- `share_instance_id` - The access that the back end grants or denies.
- `preferred` - Identifies which mount locations are most efficient and are used preferentially when multiple mount locations exist.

Data Source: flexibleengine_vbs_backup_policy_v2

The VBS Backup Policy data source provides details about a specific VBS backup policy.

Example Usage

```
variable "policy_name" { }

variable "policy_id" { }

data "flexibleengine_vbs_backup_policy_v2" "policies" {
  name = "${var.policy_name}"
  id = "${var.policy_id}"
}
```

Argument Reference

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

- `id` (Optional) - The ID of the specific VBS backup policy to retrieve.
- `name` (Optional) - The name of the specific VBS backup policy to retrieve.
- `status` (Optional) - The status of the specific VBS backup policy to retrieve. The values can be ON or OFF

Attributes Reference

The following attributes are exported:

- `id` - See Argument Reference above.
- `name` - See Argument Reference above.
- `status` - See Argument Reference above.
- `start_time` - Specifies the start time of the backup job. The value is in the HH:mm format.
- `retain_first_backup` - Specifies whether to retain the first backup in the current month.
- `retention_num` - Specifies number of retained backups.
- `frequency` - Specifies the backup interval. The value is in the range of 1 to 14 days.
- `policy_resource_count` - Specifies the number of volumes associated with the backup policy.

Data Source: flexibleengine_vbs_backup_v2

The VBS Backup data source provides details about a specific VBS Backup.

Example Usage

```
variable "backup_id" {}

data "flexibleengine_vbs_backup_v2" "mybackup" {
  id = "${var.backup_id}"
}
```

Argument Reference

The following arguments are supported:

- `id` - (Optional) The id of the vbs backup.
- `name` - (Optional) The name of the vbs backup.
- `volume_id` - (Optional) The source volume ID of the backup.
- `snapshot_id` - (Optional) ID of the snapshot associated with the backup.
- `status` - (Optional) The status of the VBS backup.

Attributes Reference

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

- `description` - The description of the vbs backup.
- `availability_zone` - The AZ where the backup resides.
- `size` - The size of the vbs backup.
- `container` - The container of the backup.
- `service_metadata` - The metadata of the vbs backup.

Data Source: flexibleengine_vpc_peering_connection_v2

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

Example Usage

```
data "flexibleengine_vpc_peering_connection_v2" "peering" {
  vpc_id      = "${flexibleengine_vpc_v1.vpc.id}"
  peer_vpc_id = "${flexibleengine_vpc_v1.peer_vpc.id}"
}

resource "flexibleengine_vpc_route_v2" "vpc_route" {
  type      = "peering"
  nexthop   = "${data.flexibleengine_vpc_peering_connection_v2.peering.id}"
  destination = "192.168.0.0/16"
  vpc_id    = "${flexibleengine_vpc_v1.vpc.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.
- `peer_vpc_id` (Optional) - The ID of the acceptor/peer VPC of the specific VPC Peering Connection to retrieve.
- `peer_tenant_id` (Optional) - The Tenant ID of the acceptor/peer VPC of the specific VPC Peering Connection to retrieve.
- `name` (Optional) - The name of the specific VPC Peering Connection to retrieve.

Attributes Reference

All of the argument attributes are exported as result attributes.

Data Source: flexibleengine_vpc_route_ids_v2

flexibleengine_vpc_route_ids_v2 provides a list of route ids for a vpc_id.

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

Example Usage

```
variable "vpc_id" { }
```

```
data "flexibleengine_vpc_route_ids_v2" "example" {  
  vpc_id = "${var.vpc_id}"  
}
```

```
data "flexibleengine_vpc_route_v2" "vpc_route" {  
  count = "${length(data.flexibleengine_vpc_route_ids_v2.example.ids)}"  
  id = "${data.flexibleengine_vpc_route_ids_v2.example.ids[count.index]}"  
}
```

```
output "route_nextHop" {  
  value = ["${data.flexibleengine_vpc_route_v2.vpc_route.*.nextHop}"]  
}
```

Argument Reference

- `vpc_id` (Required) - The VPC ID that you want to filter from.

Attributes Reference

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

Data Source: flexibleengine_vpc_route_v2

flexibleengine_vpc_route_v2 provides details about a specific VPC route.

Example Usage

```
variable "route_id" { }

data "flexibleengine_vpc_route_v2" "vpc_route" {
  id = "${var.route_id}"
}

resource "flexibleengine_vpc_subnet_v1" "subnet_v1" {
  name = "test-subnet"
  cidr = "192.168.0.0/24"
  gateway_ip = "192.168.0.1"
  vpc_id = "${data.flexibleengine_vpc_route_v2.vpc_route.vpc_id}"
}
```

Argument Reference

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

- `id` (Optional) - The id of the specific route to retrieve.
- `vpc_id` (Optional) - The id of the VPC that the desired route belongs to.
- `destination` (Optional) - The route destination address (CIDR).
- `tenant_id` (Optional) - Only the administrator can specify the tenant ID of other tenants.
- `type` (Optional) - Route type for filtering.

Attribute Reference

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

- `nexthop` - The next hop of the route. If the route type is peering, it will provide VPC peering connection ID.

Data Source: flexibleengine_vpc_subnet_ids_v1

flexibleengine_vpc_subnet_ids_v1 provides a list of subnet 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 example shows outputting all cidr blocks for every subnet id in a vpc.

```
data "flexibleengine_vpc_subnet_ids_v1" "subnet_ids" {
  vpc_id = "${var.vpc_id}"
}

data "flexibleengine_vpc_subnet_v1" "subnet" {
  count = "${length(data.flexibleengine_vpc_subnet_ids_v1.subnet_ids.ids)}"
  id     = "${data.flexibleengine_vpc_subnet_ids_v1.subnet_ids.ids[count.index]}"
}

output "subnet_cidr_blocks" {
  value = "${data.flexibleengine_vpc_subnet_v1.subnet.*.cidr}"
}
```

Argument Reference

The following arguments are supported:

- `vpc_id` (Required) - Specifies the VPC ID used as the query filter.

Attributes Reference

The following attributes are exported:

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

Data Source: flexibleengine_vpc_subnet_v1

`flexibleengine_vpc_subnet_v1` 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

```
data "flexibleengine_vpc_subnet_v1" "subnet_v1" {
  id = "${var.subnet_id}"
}

output "subnet_vpc_id" {
  value = "${data.flexibleengine_vpc_subnet_v1.subnet_v1.vpc_id}"
}
```

Argument Reference

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

- `id` - (Optional) - Specifies a resource ID in UUID format.
- `name` (Optional) - The name of the specific subnet to retrieve.
- `cidr` (Optional) - The network segment of specific subnet to retrieve. The value must be in CIDR format.
- `status` (Optional) - The value can be ACTIVE, DOWN, UNKNOWN, or ERROR.
- `vpc_id` (Optional) - The id of the VPC that the desired subnet belongs to.
- `gateway_ip` (Optional) - The subnet gateway address of specific subnet.
- `primary_dns` (Optional) - The IP address of DNS server 1 on the specific subnet.
- `secondary_dns` (Optional) - The IP address of DNS server 2 on the specific subnet.
- `availability_zone` (Optional) - The availability zone (AZ) to which the subnet should belong.

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 subnet.

- `dns_list` - The IP address list of DNS servers on the subnet.
- `dhcp_enable` - DHCP function for the subnet.

- `subnet_id` - Specifies the subnet (Native OpenStack API) ID.

flexibleengine_vpc_v1

flexibleengine_vpc_v1 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_name" {}

data "flexibleengine_vpc_v1" "vpc" {
  name = "${var.vpc_name}"
}
```

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.

- `region` - (Optional) The region in which to obtain the V1 VPC client. A VPC client is needed to retrieve VPCs. If omitted, the region argument of the provider is used.
- `id` - (Optional) The id of the specific VPC to retrieve.
- `status` - (Optional) The current status of the desired VPC. Can be either `CREATING`, `OK`, `DOWN`, `PENDING_UPDATE`, `PENDING_DELETE`, or `ERROR`.
- `name` - (Optional) A unique name for the VPC. The name must be unique for a tenant. The value is a string of no more than 64 characters and can contain digits, letters, underscores (`_`), and hyphens (`-`).
- `cidr` - (Optional) The cidr block of the desired VPC.

Attributes Reference

The following attributes are exported:

- `id` - ID of the VPC.
- `name` - See Argument Reference above.
- `status` - See Argument Reference above.

- `cidr` - See Argument Reference above.
- `routes` - The list of route information with destination and nexthop fields.
- `shared` - Specifies whether the cross-tenant sharing is supported.
- `region` - See Argument Reference above.

flexibleengine_antiddos_v1

Anti-DDoS monitors the service traffic from the Internet to ECSs, ELB instances, and BMSs to detect attack traffic in real time. It then cleans attack traffic according to user-configured defense policies so that services run as normal.

Example Usage

```
variable "eip_id" {}

resource "flexibleengine_antiddos_v1" "myantiddos" {
  floating_ip_id = "${var.eip_id}"
  enable_l7 = true
  traffic_pos_id = 1
  http_request_pos_id = 3
  cleaning_access_pos_id = 2
  app_type_id = 0
}
```

Argument Reference

The following arguments are supported:

- `enable_l7` - (Required) Specifies whether to enable L7 defense.
- `traffic_pos_id` - (Required) The position ID of traffic. The value ranges from 1 to 9.
- `http_request_pos_id` - (Required) The position ID of number of HTTP requests. The value ranges from 1 to 15.
- `cleaning_access_pos_id` - (Required) The position ID of access limit during cleaning. The value ranges from 1 to 8.
- `app_type_id` - (Required) The application type ID.
- `floating_ip_id` - (Required) The ID corresponding to the Elastic IP Address (EIP) of a user.

Attributes Reference

All above argument parameters can be exported as attribute parameters.

Import

Antiddos can be imported using the `floating_ip_id`, e.g.

```
$ terraform import flexibleengine_antiddos_v1.myantiddos c1881895-cdcb-4d23-96cb-032e6a3ee667
```

flexibleengine_as_configuration_v1

Manages a V1 AS Configuration resource within flexibleengine.

Example Usage

Basic AS Configuration

```
resource "flexibleengine_as_configuration_v1" "my_as_config" {
  scaling_configuration_name = "my_as_config"
  instance_config {
    flavor = "${var.flavor}"
    image = "${var.image_id}"
    disk {
      size = 40
      volume_type = "SATA"
      disk_type = "SYS"
    }
    key_name = "${var.keyname}"
    user_data = "${file("userdata.txt")}"
  }
}
```

AS Configuration With User Data and Metadata

```
resource "flexibleengine_as_configuration_v1" "my_as_config" {
  scaling_configuration_name = "my_as_config"
  instance_config {
    flavor = "${var.flavor}"
    image = "${var.image_id}"
    disk {
      size = 40
      volume_type = "SATA"
      disk_type = "SYS"
    }
    key_name = "${var.keyname}"
    user_data = "${file("userdata.txt")}"
    metadata = {
      some_key = "some_value"
    }
  }
}
```

`user_data` can come from a variety of sources: inline, read in from the `file` function, or the `template_cloudinit_config` resource.

AS Configuration uses the existing instance specifications as the template

```
resource "flexibleengine_as_configuration_v1" "my_as_config" {
  scaling_configuration_name = "my_as_config"
  instance_config = {
    instance_id = "4579f2f5-cbe8-425a-8f32-53dcb9d9053a"
    key_name = "${var.keyname}"
  }
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the AS configuration. If omitted, the `region` argument of the provider is used. Changing this creates a new AS configuration.
- `scaling_configuration_name` - (Required) The name of the AS configuration. The name can contain letters, digits, underscores(_), and hyphens(-), and cannot exceed 64 characters.
- `instance_config` - (Required) The information about instance configurations. The `instance_config` dictionary data structure is documented below.

The `instance_config` block supports:

- `instance_id` - (Optional) When using the existing instance specifications as the template to create AS configurations, specify this argument. In this case, `flavor`, `image`, and `disk` arguments do not take effect. If the `instance_id` argument is not specified, `flavor`, `image`, and `disk` arguments are mandatory.
- `flavor` - (Optional) The flavor ID.
- `image` - (Optional) The image ID.
- `disk` - (Optional) The disk group information. System disks are mandatory and data disks are optional. The `disk` structure is described below.
- `key_name` - (Required) The name of the SSH key pair used to log in to the instance.
- `user_data` - (Optional) The user data to provide when launching the instance. The file content must be encoded with Base64.
- `personality` - (Optional) Customize the personality of an instance by defining one or more files and their contents. The personality structure is described below.
- `public_ip` - (Optional) The elastic IP address of the instance. The `public_ip` structure is described below.
- `metadata` - (Optional) Metadata key/value pairs to make available from within the instance.

The `disk` block supports:

- `size` - (Required) The disk size. The unit is GB. The system disk size ranges from 1 to 32768, and the data disk size ranges from 10 to 32768.

- `volume_type` - (Required) The disk type, which must be the same as the disk type available in the system. The options include `SATA` (common I/O disk type) and `SSD` (ultra-high I/O disk type).
- `disk_type` - (Required) Whether the disk is a system disk or a data disk. Option `DATA` indicates a data disk. option `SYS` indicates a system disk.

The `personality` block supports:

- `path` - (Required) The absolute path of the destination file.
- `contents` - (Required) The content of the injected file, which must be encoded with base64.

The `public_ip` block supports:

- `eip` - (Required) The configuration parameter for creating an elastic IP address that will be automatically assigned to the instance. The `eip` structure is described below.

The `eip` block supports:

- `ip_type` - (Required) The IP address type. The system only supports `5_bgp` (indicates dynamic BGP).
- `bandwidth` - (Required) The bandwidth information. The structure is described below.

The `bandwidth` block supports:

- `size` - (Required) The bandwidth (Mbit/s). The value range is 1 to 300.
- `share_type` - (Required) The bandwidth sharing type. The system only supports `PER` (indicates exclusive bandwidth).
- `charging_mode` - (Required) The bandwidth charging mode. The system only supports `traffic`.

flexibleengine_as_group_v1

Manages a V1 Autoscaling Group resource within flexibleengine.

Example Usage

Basic Autoscaling Group

```
resource "flexibleengine_as_group_v1" "my_as_group" {
  scaling_group_name = "my_as_group"
  scaling_configuration_id = "37e310f5-db9d-446e-9135-c625f9c2bbfc"
  desire_instance_number = 2
  min_instance_number = 0
  max_instance_number = 10
  networks {
    id = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  }
  security_groups {
    id = "45e4c6de-6bf0-4843-8953-2babde3d4810"
  }
  vpc_id = "1d8f7e7c-fe04-4cf5-85ac-08b478c290e9"
  delete_publicip = true
  delete_instances = "yes"
}
```

Autoscaling Group Only Remove Members When Scaling Down

```
resource "flexibleengine_as_group_v1" "my_as_group_only_remove_members" {
  scaling_group_name = "my_as_group_only_remove_members"
  scaling_configuration_id = "37e310f5-db9d-446e-9135-c625f9c2bbfc"
  desire_instance_number = 2
  min_instance_number = 0
  max_instance_number = 10
  networks {
    id = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  }
  security_groups {
    id = "45e4c6de-6bf0-4843-8953-2babde3d4810"
  }
  vpc_id = "1d8f7e7c-fe04-4cf5-85ac-08b478c290e9"
  delete_publicip = true
  delete_instances = "no"
}
```

Autoscaling Group With ELB Listener

```

resource "flexibleengine_as_group_v1" "my_as_group_with_elb" {
  scaling_group_name = "my_as_group_with_elb"
  scaling_configuration_id = "37e310f5-db9d-446e-9135-c625f9c2bbfc"
  desire_instance_number = 2
  min_instance_number = 0
  max_instance_number = 10
  networks {
    id = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  }
  security_groups {
    id = "45e4c6de-6bf0-4843-8953-2babde3d4810"
  }
  vpc_id = "1d8f7e7c-fe04-4cf5-85ac-08b478c290e9"
  lb_listener_id = "${flexibleengine_elb_listener.my_listener.id}"
  delete_publicip = true
  delete_instances = "yes"
}

resource "flexibleengine_elb_listener" "my_listener" {
  name = "my_listener"
  description = "my test listener"
  protocol = "TCP"
  backend_protocol = "TCP"
  port = 12345
  backend_port = 21345
  lb_algorithm = "roundrobin"
  loadbalancer_id = "cba48790-baf5-4446-adb3-02069a916e97"
  timeouts {
    create = "5m"
    update = "5m"
    delete = "5m"
  }
}

```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the AS group. If omitted, the `region` argument of the provider is used. Changing this creates a new AS group.
- `scaling_group_name` - (Required) The name of the scaling group. The name can contain letters, digits, underscores(_), and hyphens(-), and cannot exceed 64 characters.
- `scaling_configuration_id` - (Optional) The configuration ID which defines configurations of instances in the AS group.
- `desire_instance_number` - (Optional) The expected number of instances. The default value is the minimum number of instances. The value ranges from the minimum number of instances to the maximum number of instances.
- `min_instance_number` - (Optional) The minimum number of instances. The default value is 0.
- `max_instance_number` - (Optional) The maximum number of instances. The default value is 0.

- `cool_down_time` - (Optional) The cooling duration (in seconds). The value ranges from 0 to 86400, and is 900 by default.
- `lb_listener_id` - (Optional) The ELB listener IDs. The system supports up to three ELB listeners, the IDs of which are separated using a comma (,).
- `lbaas_listeners` - (Optional) An array of one or more enhanced load balancer. The system supports the binding of up to three load balancers. The field is alternative to `lb_listener_id`. The `lbaas_listeners` object structure is documented below.
- `available_zones` - (Optional) The availability zones in which to create the instances in the autoscaling group.
- `networks` - (Required) An array of one or more network IDs. The system supports up to five networks. The `networks` object structure is documented below.
- `security_groups` - (Required) An array of one security group ID to associate with the group. The `security_groups` object structure is documented below.
- `vpc_id` - (Required) The VPC ID. Changing this creates a new group.
- `health_periodic_audit_method` - (Optional) The health check method for instances in the AS group. The health check methods include `ELB_AUDIT` and `NOVA_AUDIT`. If load balancing is configured, the default value of this parameter is `ELB_AUDIT`. Otherwise, the default value is `NOVA_AUDIT`.
- `health_periodic_audit_time` - (Optional) The health check period for instances. The period has four options: 5 minutes (default), 15 minutes, 60 minutes, and 180 minutes.
- `instance_terminate_policy` - (Optional) The instance removal policy. The policy has four options: `OLD_CONFIG_OLD_INSTANCE` (default), `OLD_CONFIG_NEW_INSTANCE`, `OLD_INSTANCE`, and `NEW_INSTANCE`.
- `notifications` - (Optional) The notification mode. The system only supports `EMAIL` mode which refers to notification by email.
- `delete_publicip` - (Optional) Whether to delete the elastic IP address bound to the instances of AS group when deleting the instances. The options are `true` and `false`.
- `delete_instances` - (Optional) Whether to delete the instances in the AS group when deleting the AS group. The options are `yes` and `no`.

The `networks` block supports:

- `id` - (Required) The network UUID.

The `security_groups` block supports:

- `id` - (Required) The UUID of the security group.

The `lbaas_listeners` block supports:

- `pool_id` - (Required) Specifies the backend ECS group ID.
- `protocol_port` - (Required) Specifies the backend protocol, which is the port on which a backend ECS listens for traffic. The number of the port ranges from 1 to 65535.
- `weight` - (Optional) Specifies the weight, which determines the portion of requests a backend ECS processes compared to other backend ECSs added to the same listener. The value of this parameter ranges from 0 to 100. The default value is 1.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `scaling_group_name` - See Argument Reference above.
- `desired_instance_number` - See Argument Reference above.
- `min_instance_number` - See Argument Reference above.
- `max_instance_number` - See Argument Reference above.
- `cool_down_time` - See Argument Reference above.
- `lb_listener_id` - See Argument Reference above.
- `health_periodic_audit_method` - See Argument Reference above.
- `health_periodic_audit_time` - See Argument Reference above.
- `instance_terminate_policy` - See Argument Reference above.
- `scaling_configuration_id` - See Argument Reference above.
- `delete_publicip` - See Argument Reference above.
- `notifications` - See Argument Reference above.
- `instances` - The instances IDs of the AS group.

flexibleengine_as_policy_v1

Manages a V1 AS Policy resource within flexibleengine.

Example Usage

AS Recurrence Policy

```
resource "flexibleengine_as_policy_v1" "hth_aspolicy"{
  scaling_policy_name = "hth_aspolicy"
  scaling_group_id = "4579f2f5-cbe8-425a-8f32-53dcb9d9053a"
  cool_down_time = 900
  scaling_policy_type = "RECURRENT"
  scaling_policy_action {
    operation = "ADD"
    instance_number = 1
  }
  scheduled_policy {
    launch_time = "07:00"
    recurrence_type = "Daily"
    start_time = "2017-11-30T12:00Z"
    end_time = "2017-12-30T12:00Z"
  }
}
```

AS Scheduled Policy

```
resource "flexibleengine_as_policy_v1" "hth_aspolicy_1"{
  scaling_policy_name = "hth_aspolicy_1"
  scaling_group_id = "4579f2f5-cbe8-425a-8f32-53dcb9d9053a"
  cool_down_time = 900
  scaling_policy_type = "SCHEDULED"
  scaling_policy_action {
    operation = "REMOVE"
    instance_number = 1
  }
  scheduled_policy {
    launch_time = "2017-12-22T12:00Z"
  }
}
```

Please note that the `launch_time` of the `SCHEDULED` policy cannot be earlier than the current time.

AS Alarm Policy

```
resource "flexibleengine_as_policy_v1" "hth_aspolicy_2"{
  scaling_policy_name = "hth_aspolicy_2"
  scaling_group_id = "4579f2f5-cbe8-425a-8f32-53dcb9d9053a"
  cool_down_time = 900
  scaling_policy_type = "ALARM"
  alarm_id = "37e310f5-db9d-446e-9135-c625f9c2bbfc"
  scaling_policy_action {
    operation = "ADD"
    instance_number = 1
  }
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the AS policy. If omitted, the `region` argument of the provider is used. Changing this creates a new AS policy.
- `scaling_policy_name` - (Required) The name of the AS policy. The name can contain letters, digits, underscores(_), and hyphens(-), and cannot exceed 64 characters.
- `scaling_group_id` - (Required) The AS group ID. Changing this creates a new AS policy.
- `scaling_policy_type` - (Required) The AS policy type. The values can be `ALARM`, `SCHEDULED`, and `RECURRENCE`.
- `alarm_id` - (Optional) The alarm rule ID. This argument is mandatory when `scaling_policy_type` is set to `ALARM`.
- `scheduled_policy` - (Optional) The periodic or scheduled AS policy. This argument is mandatory when `scaling_policy_type` is set to `SCHEDULED` or `RECURRENCE`. The `scheduled_policy` structure is documented below.
- `scaling_policy_action` - (Optional) The action of the AS policy. The `scaling_policy_action` structure is documented below.
- `cool_down_time` - (Optional) The cooling duration (in seconds), and is 900 by default.

The `scheduled_policy` block supports:

- `launch_time` - (Required) The time when the scaling action is triggered. If `scaling_policy_type` is set to `SCHEDULED`, the time format is `YYYY-MM-DDThh:mmZ`. If `scaling_policy_type` is set to `RECURRENCE`, the time format is `hh:mm`.
- `recurrence_type` - (Optional) The periodic triggering type. This argument is mandatory when `scaling_policy_type` is set to `RECURRENCE`. The options include `Daily`, `Weekly`, and `Monthly`.
- `recurrence_value` - (Optional) The frequency at which scaling actions are triggered.
- `start_time` - (Optional) The start time of the scaling action triggered periodically. The time format complies with UTC. The current time is used by default. The time format is `YYYY-MM-DDThh:mmZ`.
- `end_time` - (Optional) The end time of the scaling action triggered periodically. The time format complies with UTC. This argument is mandatory when `scaling_policy_type` is set to `RECURRENCE`. The time format is `YYYY-MM-DDThh:mmZ`.

The `scaling_policy_action` block supports:

- `operation` - (Optional) The operation to be performed. The options include `ADD` (default), `REMOVE`, and `SET`.
- `instance_number` - (Optional) The number of instances to be operated. The default number is 1.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `scaling_policy_name` - See Argument Reference above.
- `scaling_policy_type` - See Argument Reference above.
- `alarm_id` - See Argument Reference above.
- `cool_down_time` - See Argument Reference above.
- `scaling_policy_action/operation` - See Argument Reference above.
- `scaling_policy_action/instance_number` - See Argument Reference above.
- `scheduled_policy/launch_time` - See Argument Reference above.
- `scheduled_policy/recurrence_type` - See Argument Reference above.
- `scheduled_policy/recurrence_value` - See Argument Reference above.
- `scheduled_policy/start_time` - See Argument Reference above.
- `scheduled_policy/end_time` - See Argument Reference above.

flexibleengine_blockstorage_volume_v2

Manages a V2 volume resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_blockstorage_volume_v2" "volume_1" {
  region      = "RegionOne"
  name        = "volume_1"
  description = "first test volume"
  size        = 3
  metadata = {
    __system__encrypted = "1"
    __system__cmkid     = "kms_id"
  }
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the volume. If omitted, the `region` argument of the provider is used. Changing this creates a new volume.
- `size` - (Required) The size of the volume to create (in gigabytes).
- `availability_zone` - (Optional) The availability zone for the volume. Changing this creates a new volume.
- `consistency_group_id` - (Optional) The consistency group to place the volume in.
- `description` - (Optional) A description of the volume. Changing this updates the volume's description.
- `image_id` - (Optional) The image ID from which to create the volume. Changing this creates a new volume.
- `metadata` - (Optional) Metadata key/value pairs to associate with the volume. Changing this updates the existing volume metadata. The EVS encryption capability with KMS key can be set with the following parameters:
 - `__system__encrypted` - The default value is set to '0', which means the volume is not encrypted, the value '1' indicates volume is encrypted.
 - `__system__cmkid` - (Optional) The ID of the kms key.
- `name` - (Optional) A unique name for the volume. Changing this updates the volume's name.
- `snapshot_id` - (Optional) The snapshot ID from which to create the volume. Changing this creates a new volume.
- `source_replica` - (Optional) The volume ID to replicate with.
- `source_vol_id` - (Optional) The volume ID from which to create the volume. Changing this creates a new volume.
- `volume_type` - (Optional) The type of volume to create. Changing this creates a new volume.

- `cascade` - (Optional, Default:false) Specifies to delete all snapshots associated with the EVS disk.
- `multiattach` - (Optional) Specifies whether the EVS disk is shareable.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `size` - See Argument Reference above.
- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `availability_zone` - See Argument Reference above.
- `image_id` - See Argument Reference above.
- `source_vol_id` - See Argument Reference above.
- `snapshot_id` - See Argument Reference above.
- `metadata` - See Argument Reference above.
- `volume_type` - See Argument Reference above.
- `multiattach` - See Argument Reference above.
- `attachment` - If a volume is attached to an instance, this attribute will display the Attachment ID, Instance ID, and the Device as the Instance sees it.

Import

Volumes can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_blockstorage_volume_v2.volume_1 ea257959-eeb1-4c10-8d33-26f0409a755d
```

flexibleengine_cce_cluster_v3

Provides a cluster resource (CCE).

Example Usage

```
variable "flavor_id" { }
variable "vpc_id" { }
variable "subnet_id" { }

resource "flexibleengine_cce_cluster_v3" "cluster_1" {
  name = "cluster"
  cluster_type= "VirtualMachine"
  flavor_id= "${var.flavor_id}"
  vpc_id= "${var.vpc_id}"
  subnet_id= "${var.subnet_id}"
  container_network_type= "overlay_l2"
  authentication_mode = "rbac"
  description= "Create cluster"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) Cluster name. Changing this parameter will create a new cluster resource.
- `labels` - (Optional) Cluster tag, key/value pair format. Changing this parameter will create a new cluster resource.
- `annotations` - (Optional) Cluster annotation, key/value pair format. Changing this parameter will create a new cluster resource.
- `flavor_id` - (Required) Cluster specifications. Changing this parameter will create a new cluster resource.
 - `cce.s1.small` - small-scale single cluster (up to 50 nodes).
 - `cce.s1.medium` - medium-scale single cluster (up to 200 nodes).
 - `cce.s1.large` - large-scale single cluster (up to 1000 nodes).
 - `cce.s2.small` - small-scale HA cluster (up to 50 nodes).
 - `cce.s2.medium` - medium-scale HA cluster (up to 200 nodes).
 - `cce.s2.large` - large-scale HA cluster (up to 1000 nodes).
 - `cce.t1.small` - small-scale single physical machine cluster (up to 10 nodes).
 - `cce.t1.medium` - medium-scale single physical machine cluster (up to 100 nodes).
 - `cce.t1.large` - large-scale single physical machine cluster (up to 500 nodes).

- `cce.t2.small` - small-scale HA physical machine cluster (up to 10 nodes).
- `cce.t2.medium` - medium-scale HA physical machine cluster (up to 100 nodes).
- `cce.t2.large` - large-scale HA physical machine cluster (up to 500 nodes).
- `cluster_version` - (Optional) For the cluster version, possible value is v1.9.7-r1.
- `cluster_type` - (Required) Cluster Type, possible values are VirtualMachine and BareMetal. Changing this parameter will create a new cluster resource.
- `description` - (Optional) Cluster description.
- `billing_mode` - (Optional) Charging mode of the cluster, which is 0 (on demand). Changing this parameter will create a new cluster resource.
- `extend_param` - (Optional) Extended parameter. Changing this parameter will create a new cluster resource.
- `vpc_id` - (Required) The ID of the VPC used to create the node. Changing this parameter will create a new cluster resource.
- `subnet_id` - (Required) The NETWORK ID of the subnet used to create the node. Changing this parameter will create a new cluster resource.
- `highway_subnet_id` - (Optional) The ID of the high speed network used to create bare metal nodes. Changing this parameter will create a new cluster resource.
- `container_network_type` - (Required) Container network parameters. Possible values:
 - `overlay_l2` - An overlay_l2 network built for containers by using Open vSwitch(OVS)
 - `underlay_ipvlan` - An underlay_ipvlan network built for bare metal servers by using ipvlan.
 - `vpc-router` - An vpc-router network built for containers by using ipvlan and custom VPC routes.
- `container_network_cidr` - (Optional) Container network segment. Changing this parameter will create a new cluster resource.
- `authentication_mode` - (Optional) Authentication mode of the cluster, possible values are x509 and rbac. Defaults to x509. Changing this parameter will create a new cluster resource.

Attributes Reference

All above argument parameters can be exported as attribute parameters along with attribute reference.

- `id` - Id of the cluster resource.
- `status` - Cluster status information.
- `internal_endpoint` - The internal network address.
- `external_endpoint` - The external network address.
- `external_apig_endpoint` - The endpoint of the cluster to be accessed through API Gateway.
- `security_group_id` - Security group ID of the cluster.

Import

Cluster can be imported using the cluster id, e.g.

```
$ terraform import flexibleengine_cce_cluster_v3.cluster_1 4779ab1c-7c1a-44b1-a02e-93dfc361b32d
```

flexibleengine_cce_nodes_v3

Add a node to a container cluster.

Example Usage

```
variable "cluster_id" { }
variable "ssh_key" { }
variable "availability_zone" { }

resource "flexibleengine_cce_node_v3" "node_1" {
  cluster_id="${var.cluster_id}"
  name = "node1"
  flavor_id="s1.medium"
  iptype="5_bgp"
  availability_zone= "${var.availability_zone}"
  key_pair="${var.ssh_key}"
  root_volume {
    size= 40
    volumetype= "SATA"
  }
  sharetype= "PER"
  bandwidth_size= 100
  data_volumes {
    size= 100
    volumetype= "SATA"
  }
}
```

Argument Reference

The following arguments are supported:

- `cluster_id` - (Required) ID of the cluster. Changing this parameter will create a new resource.
- `billing_mode` - (Optional) Node's billing mode: The value is 0 (on demand). Changing this parameter will create a new resource.
- `name` - (Optional) Node Name.
- `labels` - (Optional) Node tag, key/value pair format. Changing this parameter will create a new resource.
- `annotations` - (Optional) Node annotation, key/value pair format. Changing this parameter will create a new resource.
- `flavor_id` - (Required) Specifies the flavor id. Changing this parameter will create a new resource.
- `availability_zone` - (Required) specify the name of the available partition (AZ). Changing this parameter will create a new resource.

- `os` - (Optional) Operating System of the node, possible values are EulerOS 2.2 and CentOS 7.6. Defaults to EulerOS 2.2. Changing this parameter will create a new resource.
- `key_pair` - (Required) Key pair name when logging in to select the key pair mode. Changing this parameter will create a new resource.
- `eip_ids` - (Optional) List of existing elastic IP IDs. Changing this parameter will create a new resource.

Note: If the `eip_ids` parameter is configured, you do not need to configure the `eip_count` and `bandwidth` parameters: `iptype`, `charge_mode`, `bandwidth_size` and `share_type`.

- `eip_count` - (Optional) Number of elastic IPs to be dynamically created. Changing this parameter will create a new resource.
- `iptype` - (Required) Elastic IP type.
- `bandwidth_charge_mode` - (Optional) Bandwidth billing type. Changing this parameter will create a new resource.
- `sharetype` - (Required) Bandwidth sharing type. Changing this parameter will create a new resource.
- `bandwidth_size` - (Required) Bandwidth size. Changing this parameter will create a new resource.
- `extend_param_charging_mode` - (Optional) Node charging mode, 0 is on-demand charging. Changing this parameter will create a new cluster resource.
- `ecs_performance_type` - (Optional) Classification of cloud server specifications. Changing this parameter will create a new cluster resource.
- `order_id` - (Optional) Order ID, mandatory when the node payment type is the automatic payment package period type. Changing this parameter will create a new cluster resource.
- `product_id` - (Optional) The Product ID. Changing this parameter will create a new cluster resource.
- `max_pods` - (Optional) The maximum number of instances a node is allowed to create. Changing this parameter will create a new cluster resource.
- `public_key` - (Optional) The Public key. Changing this parameter will create a new cluster resource.

root_volume - (Required) It corresponds to the system disk related configuration. Changing this parameter will create a new resource.

- `size` - (Required) Disk size in GB.
- `volumentype` - (Required) Disk type.
- `extend_param` - (Optional) Disk expansion parameters.

data_volumes - (Required) Represents the data disk to be created. Changing this parameter will create a new resource.

- `size` - (Required) Disk size in GB.
- `volumentype` - (Required) Disk type.
- `extend_param` - (Optional) Disk expansion parameters.

Attributes Reference

All above argument parameters can be exported as attribute parameters along with attribute reference.

- `status` - Node status information.
- `private_ip` - Private IP of the CCE node.
- `public_ip` - Public IP of the CCE node.

flexibleengine_ces_alarmrule

Manages a V2 topic resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_ces_alarmrule" "alarm_rule" {
  "alarm_name" = "alarm_rule"
  "metric" {
    "namespace" = "SYS.ECS"
    "metric_name" = "network_outgoing_bytes_rate_inband"
    "dimensions" {
      "name" = "instance_id"
      "value" = "${flexibleengine_compute_instance_v2.webserver.id}"
    }
  }
  "condition" {
    "period" = 300
    "filter" = "average"
    "comparison_operator" = ">"
    "value" = 6
    "unit" = "B/s"
    "count" = 1
  }
  "alarm_actions" {
    "type" = "notification"
    "notification_list" = [
      "${flexibleengine_smn_topic_v2.topic.id}"
    ]
  }
  "alarm_actions" {
    "type" = "notification"
    "notification_list" = [
      "${flexibleengine_smn_topic_v2.topic_2.id}"
    ]
  }
}
```

Argument Reference

The following arguments are supported:

- `alarm_name` - (Required) Specifies the name of an alarm rule. The value can be a string of 1 to 128 characters that can consist of numbers, lowercase letters, uppercase letters, underscores (`_`), or hyphens (`-`).
- `alarm_description` - (Optional) The value can be a string of 0 to 256 characters.
- `metric` - (Required) Specifies the alarm metrics. The structure is described below.
- `condition` - (Required) Specifies the alarm triggering condition. The structure is described below.

- `alarm_actions` - (Optional) Specifies the action list triggered by an alarm. The structure is described below.
- `insufficientdata_actions` - (Optional) Specifies the action list triggered by data insufficiency. The structure is described below.
- `ok_actions` - (Optional) Specifies the action list triggered by the clearing of an alarm. The structure is described below.
- `alarm_enabled` - (Optional) Specifies whether to enable the alarm. The default value is true.
- `alarm_action_enabled` - (Optional) Specifies whether to enable the action to be triggered by an alarm. The default value is true. Note: If `alarm_action_enabled` is set to true, at least one of the following parameters `alarm_actions`, `insufficientdata_actions`, and `ok_actions` cannot be empty. If `alarm_actions`, `insufficientdata_actions`, and `ok_actions` coexist, their corresponding `notification_list` must be of the same value.

The `metric` block supports:

- `namespace` - (Required) Specifies the namespace in `service.item` format. `service.item` can be a string of 3 to 32 characters that must start with a letter and can consist of uppercase letters, lowercase letters, numbers, or underscores (`_`).
- `metric_name` - (Required) Specifies the metric name. The value can be a string of 1 to 64 characters that must start with a letter and can consist of uppercase letters, lowercase letters, numbers, or underscores (`_`).
- `dimensions` - (Required) Specifies the list of metric dimensions. Currently, the maximum length of the dimension list that are supported is 3. The structure is described below.

The `dimensions` block supports:

- `name` - (Required) Specifies the dimension name. The value can be a string of 1 to 32 characters that must start with a letter and can consist of uppercase letters, lowercase letters, numbers, underscores (`_`), or hyphens (`-`).
- `value` - (Required) Specifies the dimension value. The value can be a string of 1 to 64 characters that must start with a letter or a number and can consist of uppercase letters, lowercase letters, numbers, underscores (`_`), or hyphens (`-`).

The `condition` block supports:

- `period` - (Required) Specifies the alarm checking period in seconds. The value can be 1, 300, 1200, 3600, 14400, and 86400. Note: If `period` is set to 1, the raw metric data is used to determine whether to generate an alarm.
- `filter` - (Required) Specifies the data rollup methods. The value can be `max`, `min`, `average`, `sum`, and `variance`.
- `comparison_operator` - (Required) Specifies the comparison condition of alarm thresholds. The value can be `>`, `=`, `<`, `>=`, or `<=`.
- `value` - (Required) Specifies the alarm threshold. The value ranges from 0 to Number of 1.7976931348623157e+308.
- `unit` - (Optional) Specifies the data unit.
- `count` - (Required) Specifies the number of consecutive occurrence times. The value ranges from 1 to 5.

the `alarm_actions` block supports:

- `type` - (Optional) specifies the type of action triggered by an alarm. the value can be `notification` or `autoscaling`. `notification`: indicates that a notification will be sent to the user. `autoscaling`: indicates that a scaling action will be triggered.

- `notification_list` - (Required) specifies the topic urn list of the target notification objects. the maximum length is 5. the topic urn list can be obtained from simple message notification (smn) and in the following format: `urn: smn:([a-z]|[a-z][0-9]|-){1,32}:([a-z]|[a-z][0-9]){32}:([a-z]|[a-z][0-9]|-|_){1,256}`. if type is set to notification, the value of `notification_list` cannot be empty. if type is set to autoscaling, the value of `notification_list` must be [] and the value of `namespace` must be `sys.as`. Note: to enable the as alarm rules take effect, you must bind scaling policies. for details, see the auto scaling api reference.

the `insufficientdata_actions` block supports:

- `type` - (Optional) specifies the type of action triggered by an alarm. the value is notification. notification: indicates that a notification will be sent to the user.
- `notification_list` - (Optional) indicates the list of objects to be notified if the alarm status changes. the maximum length is 5.

the `ok_actions` block supports:

- `type` - (Optional) specifies the type of action triggered by an alarm. the value is notification. notification: indicates that a notification will be sent to the user.
- `notification_list` - (Optional) indicates the list of objects to be notified if the alarm status changes. the maximum length is 5.

Attributes Reference

The following attributes are exported:

- `alarm_name` - See Argument Reference above.
- `alarm_description` - See Argument Reference above.
- `metric` - See Argument Reference above.
- `condition` - See Argument Reference above.
- `alarm_actions` - See Argument Reference above.
- `insufficientdata_actions` - See Argument Reference above.
- `ok_actions` - See Argument Reference above.
- `alarm_enabled` - See Argument Reference above.
- `alarm_action_enabled` - See Argument Reference above.
- `id` - Specifies the alarm rule ID.
- `update_time` - Specifies the time when the alarm status changed. The value is a UNIX timestamp and the unit is ms.
- `alarm_state` - Specifies the alarm status. The value can be: `ok`: The alarm status is normal, `alarm`: An alarm is generated, `insufficient_data`: The required data is insufficient.

flexibleengine_compute_bms_server_v2

Manages a BMS Server resource within FlexibleEngine.

Example Usage

Basic Instance

```
variable "image_id" {}
variable "flavor_id" {}
variable "keypair_name" {}
variable "network_id" {}
variable "availability_zone" {}

resource "flexibleengine_compute_bms_server_v2" "basic" {
  name          = "basic"
  image_id      = "${var.image_id}"
  flavor_id     = "${var.flavor_id}"
  key_pair      = "${var.keypair_name}"
  security_groups = ["default"]
  availability_zone = "${var.availability_zone}"

  metadata = {
    this = "that"
  }

  network {
    uuid = "${var.network_id}"
  }
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the bms server instance. If omitted, the `region` argument of the provider is used. Changing this creates a new bms server.
- `name` - (Required) The name of the BMS.
- `image_id` - (Optional; Required if `image_name` is empty.) Changing this creates a new bms server.
- `image_name` - (Optional; Required if `image_id` is empty.) The name of the desired image for the bms server. Changing this creates a new bms server.
- `flavor_id` - (Optional; Required if `flavor_name` is empty) The flavor ID of the desired flavor for the bms server. Changing this resizes the existing bms server.

- `flavor_name` - (Optional; Required if `flavor_id` is empty) The name of the desired flavor for the bms server. Changing this resizes the existing bms server.
- `user_data` - (Optional) The user data to provide when launching the instance. Changing this creates a new bms server.
- `security_groups` - (Optional) An array of one or more security group names to associate with the bms server. Changing this results in adding/removing security groups from the existing bms server.
- `availability_zone` - (Required) The availability zone in which to create the bms server.
- `network` - (Optional) An array of one or more networks to attach to the bms instance. Changing this creates a new bms server.
- `metadata` - (Optional) Metadata key/value pairs to make available from within the instance. Changing this updates the existing bms server metadata.
- `admin_pass` - (Optional) The administrative password to assign to the bms server. Changing this changes the root password on the existing server.
- `key_pair` - (Optional) The name of a key pair to put on the bms server. The key pair must already be created and associated with the tenant's account. Changing this creates a new bms server.
- `stop_before_destroy` - (Optional) Whether to try stop instance gracefully before destroying it, thus giving chance for guest OS daemons to stop correctly. If instance doesn't stop within timeout, it will be destroyed anyway.

The `network` block supports:

- `uuid` - (Required unless `port` or `name` is provided) The network UUID to attach to the bms server. Changing this creates a new bms server.
- `name` - (Required unless `uuid` or `port` is provided) The human-readable name of the network. Changing this creates a new bms server.
- `port` - (Required unless `uuid` or `name` is provided) The port UUID of a network to attach to the bms server. Changing this creates a new server.
- `fixed_ip_v4` - (Optional) Specifies a fixed IPv4 address to be used on this network. Changing this creates a new bms server.
- `fixed_ip_v6` - (Optional) Specifies a fixed IPv6 address to be used on this network. Changing this creates a new bms server.
- `access_network` - (Optional) Specifies if this network should be used for provisioning access. Accepts true or false. Defaults to false.

Attributes Reference

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

- `id` - The id of the bms server.
- `config_drive` - Whether to use the `config_drive` feature to configure the instance.
- `kernel_id` - The UUID of the kernel image when the AMI image is used.

- `user_id` - The ID of the user to which the BMS belongs.
- `host_status` - The nova-compute status: **UP**, **UNKNOWN**, **DOWN**, **MAINTENANCE** and **Null**.

flexibleengine_compute_floatingip_associate_v2

Associate a floating IP to an instance. This can be used instead of the `floating_ip` options in `flexibleengine_compute_instance_v2`.

Example Usage

Associate with EIP

```
resource "flexibleengine_compute_instance_v2" "instance_1" {
  name           = "instance_1"
  image_id      = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  flavor_id     = 3
  key_pair      = "my_key_pair_name"
  security_groups = ["default"]
}

resource "flexibleengine_vpc_eip_v1" "eip_1" {
  publicip {
    type = "5_bgp"
  }
  bandwidth {
    name = "test"
    size = 8
    share_type = "PER"
    charge_mode = "traffic"
  }
}

resource "flexibleengine_compute_floatingip_associate_v2" "fip_1" {
  floating_ip = "${flexibleengine_vpc_eip_v1.eip_1.publicip.0.ip_address}"
  instance_id = "${flexibleengine_compute_instance_v2.instance_1.id}"
}
```

Automatically detect the correct network

```

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  image_id      = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  flavor_id     = 3
  key_pair      = "my_key_pair_name"
  security_groups = ["default"]
}

resource "flexibleengine_networking_floatingip_v2" "fip_1" {
  pool = "admin_external_net"
}

resource "flexibleengine_compute_floatingip_associate_v2" "fip_1" {
  floating_ip = "${flexibleengine_networking_floatingip_v2.fip_1.address}"
  instance_id = "${flexibleengine_compute_instance_v2.instance_1.id}"
}

```

Explicitly set the network to attach to

```

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  image_id      = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  flavor_id     = 3
  key_pair      = "my_key_pair_name"
  security_groups = ["default"]

  network {
    name = "my_network"
  }

  network {
    name = "default"
  }
}

resource "flexibleengine_networking_floatingip_v2" "fip_1" {
  pool = "admin_external_net"
}

resource "flexibleengine_compute_floatingip_associate_v2" "fip_1" {
  floating_ip = "${flexibleengine_networking_floatingip_v2.fip_1.address}"
  instance_id = "${flexibleengine_compute_instance_v2.instance_1.id}"
  fixed_ip   = "${flexibleengine_compute_instance_v2.instance_1.network.1.fixed_ip_v4}"
}

```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Compute client. Keypairs are associated with accounts, but a Compute client is needed to create one. If omitted, the `region` argument of the provider is used. Changing this creates a new `floatingip_associate`.
- `floating_ip` - (Required) The floating IP to associate.
- `instance_id` - (Required) The instance to associate the floating IP with.
- `fixed_ip` - (Optional) The specific IP address to direct traffic to.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `floating_ip` - See Argument Reference above.
- `instance_id` - See Argument Reference above.
- `fixed_ip` - See Argument Reference above.

Import

This resource can be imported by specifying all three arguments, separated by a forward slash:

```
$ terraform import flexibleengine_compute_floatingip_associate_v2.fip_1 <floating_ip>/<instance_id>/<fixed_ip>
```

flexibleengine_compute_floatingip_v2

Manages a V2 floating IP resource within FlexibleEngine Nova (compute) that can be used for compute instances. These are similar to Neutron (networking) floating IP resources, but only networking floating IPs can be used with load balancers.

Example Usage

```
resource "flexibleengine_compute_floatingip_v2" "floatip_1" {  
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Compute client. A Compute client is needed to create a floating IP that can be used with a compute instance. If omitted, the `region` argument of the provider is used. Changing this creates a new floating IP (which may or may not have a different address).
- `pool` - (Optional) The name of the pool from which to obtain the floating IP. Default value is `admin_external_net`. Changing this creates a new floating IP.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `pool` - See Argument Reference above.
- `address` - The actual floating IP address itself.
- `fixed_ip` - The fixed IP address corresponding to the floating IP.
- `instance_id` - UUID of the compute instance associated with the floating IP.

Import

Floating IPs can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_compute_floatingip_v2.floatip_1 89c60255-9bd6-460c-822a-e2b959ede9d2
```

flexibleengine_compute_instance_v2

Manages a V2 VM instance resource within FlexibleEngine.

Example Usage

Basic Instance

```
resource "flexibleengine_compute_instance_v2" "basic" {
  name           = "basic"
  image_id       = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  flavor_id      = "3"
  key_pair       = "my_key_pair_name"
  security_groups = ["default"]

  metadata = {
    this = "that"
  }

  network {
    name = "my_network"
  }
}
```

Instance With Attached Volume

```
resource "flexibleengine_blockstorage_volume_v2" "myvol" {
  name = "myvol"
  size = 1
}

resource "flexibleengine_compute_instance_v2" "myinstance" {
  name           = "myinstance"
  image_id       = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  flavor_id      = "3"
  key_pair       = "my_key_pair_name"
  security_groups = ["default"]

  network {
    name = "my_network"
  }
}

resource "flexibleengine_compute_volume_attach_v2" "attached" {
  instance_id = "${flexibleengine_compute_instance_v2.myinstance.id}"
  volume_id   = "${flexibleengine_blockstorage_volume_v2.myvol.id}"
}
```

Boot From Volume

```
resource "flexibleengine_compute_instance_v2" "boot-from-volume" {
  name          = "boot-from-volume"
  flavor_id     = "3"
  key_pair      = "my_key_pair_name"
  security_groups = ["default"]

  block_device {
    uuid          = "<image-id>"
    source_type   = "image"
    volume_size   = 5
    boot_index    = 0
    destination_type = "volume"
    delete_on_termination = true
    volume_type   = "SSD"
  }

  network {
    name = "my_network"
  }
}
```

Boot From an Existing Volume

```
resource "flexibleengine_blockstorage_volume_v2" "myvol" {
  name    = "myvol"
  size    = 5
  image_id = "<image-id>"
}

resource "flexibleengine_compute_instance_v2" "boot-from-volume" {
  name          = "bootfromvolume"
  flavor_id     = "3"
  key_pair      = "my_key_pair_name"
  security_groups = ["default"]

  block_device {
    uuid          = "${flexibleengine_blockstorage_volume_v2.myvol.id}"
    source_type   = "volume"
    boot_index    = 0
    destination_type = "volume"
    delete_on_termination = true
  }

  network {
    name = "my_network"
  }
}
```

Boot Instance, Create Volume, and Attach Volume as a Block Device

```

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  image_id      = "<image-id>"
  flavor_id     = "3"
  key_pair      = "my_key_pair_name"
  security_groups = ["default"]

  block_device {
    uuid          = "<image-id>"
    source_type   = "image"
    destination_type = "local"
    boot_index    = 0
    delete_on_termination = true
  }

  block_device {
    source_type       = "blank"
    destination_type  = "volume"
    volume_size       = 1
    boot_index        = 1
    delete_on_termination = true
  }
}

```

Boot Instance and Attach Existing Volume as a Block Device

```

resource "flexibleengine_blockstorage_volume_v2" "volume_1" {
  name = "volume_1"
  size = 1
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  image_id      = "<image-id>"
  flavor_id     = "3"
  key_pair      = "my_key_pair_name"
  security_groups = ["default"]

  block_device {
    uuid          = "<image-id>"
    source_type   = "image"
    destination_type = "local"
    boot_index    = 0
    delete_on_termination = true
  }

  block_device {
    uuid          = "${flexibleengine_blockstorage_volume_v2.volume_1.id}"
    source_type   = "volume"
    destination_type = "volume"
    boot_index    = 1
    delete_on_termination = true
  }
}

```

Instance With Multiple Networks

```
resource "flexibleengine_networking_floatingip_v2" "myip" {
  pool = "admin_external_net"
}

resource "flexibleengine_compute_instance_v2" "multi-net" {
  name          = "multi-net"
  image_id      = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  flavor_id     = "3"
  key_pair      = "my_key_pair_name"
  security_groups = ["default"]

  network {
    name = "my_first_network"
  }

  network {
    name = "my_second_network"
  }
}

resource "flexibleengine_compute_floatingip_associate_v2" "myip" {
  floating_ip = "${flexibleengine_networking_floatingip_v2.myip.address}"
  instance_id = "${flexibleengine_compute_instance_v2.multi-net.id}"
  fixed_ip    = "${flexibleengine_compute_instance_v2.multi-net.network.1.fixed_ip_v4}"
}
```

Instance with Multiple Ephemeral Disks

```

resource "flexibleengine_compute_instance_v2" "multi-eph" {
  name          = "multi-eph"
  image_id      = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  flavor_id     = "3"
  key_pair      = "my_key_pair_name"
  security_groups = ["default"]

  block_device {
    boot_index          = 0
    delete_on_termination = true
    destination_type    = "local"
    source_type         = "image"
    uuid                = "<image-id>"
  }

  block_device {
    boot_index          = -1
    delete_on_termination = true
    destination_type    = "local"
    source_type         = "blank"
    volume_size         = 1
  }

  block_device {
    boot_index          = -1
    delete_on_termination = true
    destination_type    = "local"
    source_type         = "blank"
    volume_size         = 1
  }
}

```

Instance with User Data (cloud-init)

```

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "basic"
  image_id      = "ad091b52-742f-469e-8f3c-fd81cadf0743"
  flavor_id     = "3"
  key_pair      = "my_key_pair_name"
  security_groups = ["default"]
  user_data     = "#cloud-config\nhostname: instance_1.example.com\nfqdn: instance_1.example.com"

  network {
    name = "my_network"
  }
}

```

`user_data` can come from a variety of sources: inline, read in from the `file` function, or the `template_cloudinit_config` resource.

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the server instance. If omitted, the `region` argument of the provider is used. Changing this creates a new server.
- `name` - (Required) A unique name for the resource.
- `image_id` - (Optional; Required if `image_name` is empty and not booting from a volume. Do not specify if booting from a volume.) The image ID of the desired image for the server. Changing this creates a new server.
- `image_name` - (Optional; Required if `image_id` is empty and not booting from a volume. Do not specify if booting from a volume.) The name of the desired image for the server. Changing this creates a new server.
- `flavor_id` - (Optional; Required if `flavor_name` is empty) The flavor ID of the desired flavor for the server. Changing this resizes the existing server.
- `flavor_name` - (Optional; Required if `flavor_id` is empty) The name of the desired flavor for the server. Changing this resizes the existing server.
- `user_data` - (Optional) The user data to provide when launching the instance. Changing this creates a new server.
- `security_groups` - (Optional) An array of one or more security group names to associate with the server. Changing this results in adding/removing security groups from the existing server. *Note:* When attaching the instance to networks using Ports, place the security groups on the Port and not the instance.
- `availability_zone` - (Optional) The availability zone in which to create the server. Changing this creates a new server.
- `network` - (Optional) An array of one or more networks to attach to the instance. The network object structure is documented below. Changing this creates a new server.
- `metadata` - (Optional) Metadata key/value pairs to make available from within the instance. Changing this updates the existing server metadata.
- `config_drive` - (Optional) Whether to use the `config_drive` feature to configure the instance. Changing this creates a new server.
- `admin_pass` - (Optional) The administrative password to assign to the server. Changing this changes the root password on the existing server.
- `key_pair` - (Optional) The name of a key pair to put on the server. The key pair must already be created and associated with the tenant's account. Changing this creates a new server.
- `block_device` - (Optional) Configuration of block devices. The `block_device` structure is documented below. Changing this creates a new server. You can specify multiple block devices which will create an instance with multiple disks. This configuration is very flexible, so please see the following reference (http://docs.openstack.org/developer/nova/block_device_mapping.html) for more information.
- `scheduler_hints` - (Optional) Provide the Nova scheduler with hints on how the instance should be launched. The available hints are described below.
- `stop_before_destroy` - (Optional) Whether to try stop instance gracefully before destroying it, thus giving chance for guest OS daemons to stop correctly. If instance doesn't stop within timeout, it will be destroyed anyway.
- `force_delete` - (Optional) Whether to force the FlexibleEngine instance to be forcefully deleted. This is useful for environments that have reclaim / soft deletion enabled.

- `auto_recovery` - (Optional) Configures or deletes automatic recovery of an instance

The `network` block supports:

- `uuid` - (Required unless `port` or `name` is provided) The network UUID to attach to the server. Changing this creates a new server.
- `name` - (Required unless `uuid` or `port` is provided) The human-readable name of the network. Changing this creates a new server.
- `port` - (Required unless `uuid` or `name` is provided) The port UUID of a network to attach to the server. Changing this creates a new server.
- `fixed_ip_v4` - (Optional) Specifies a fixed IPv4 address to be used on this network. Changing this creates a new server.
- `fixed_ip_v6` - (Optional) Specifies a fixed IPv6 address to be used on this network. Changing this creates a new server.
- `access_network` - (Optional) Specifies if this network should be used for provisioning access. Accepts true or false. Defaults to false.

The `block_device` block supports:

- `uuid` - (Required unless `source_type` is set to "blank") The UUID of the image, volume, or snapshot. Changing this creates a new server.
- `source_type` - (Required) The source type of the device. Must be one of "blank", "image", "volume", or "snapshot". Changing this creates a new server.
- `volume_size` - The size of the volume to create (in gigabytes). Required in the following combinations: `source=image` and `destination=volume`, `source=blank` and `destination=local`, and `source=blank` and `destination=volume`. Changing this creates a new server.
- `volume_type` - (Optional) Currently, the value can be `SSD` (ultra-I/O disk type), `SAS` (high I/O disk type), or `SATA` (common I/O disk type)
- `boot_index` - (Optional) The boot index of the volume. It defaults to 0, which indicates that it's a system disk. Changing this creates a new server.
- `destination_type` - (Optional) The type that gets created. Possible values are "volume" and "local". Changing this creates a new server.
- `delete_on_termination` - (Optional) Delete the volume / block device upon termination of the instance. Defaults to false. Changing this creates a new server.

The `scheduler_hints` block supports:

- `group` - (Optional) A UUID of a Server Group. The instance will be placed into that group.
- `different_host` - (Optional) A list of instance UUIDs. The instance will be scheduled on a different host than all other instances.
- `same_host` - (Optional) A list of instance UUIDs. The instance will be scheduled on the same host of those specified.
- `query` - (Optional) A conditional query that a compute node must pass in order to host an instance.

- `target_cell` - (Optional) The name of a cell to host the instance.
- `build_near_host_ip` - (Optional) An IP Address in CIDR form. The instance will be placed on a compute node that is in the same subnet.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `access_ip_v4` - The first detected Fixed IPv4 address *or* the Floating IP.
- `access_ip_v6` - The first detected Fixed IPv6 address.
- `metadata` - See Argument Reference above.
- `security_groups` - See Argument Reference above.
- `flavor_id` - See Argument Reference above.
- `flavor_name` - See Argument Reference above.
- `network/uuid` - See Argument Reference above.
- `network/name` - See Argument Reference above.
- `network/port` - See Argument Reference above.
- `network/fixed_ip_v4` - The Fixed IPv4 address of the Instance on that network.
- `network/fixed_ip_v6` - The Fixed IPv6 address of the Instance on that network.
- `network/mac` - The MAC address of the NIC on that network.
- `all_metadata` - Contains all instance metadata, even metadata not set by Terraform.
- `auto_recovery` - See Argument Reference above.

Notes

Multiple Ephemeral Disks

It's possible to specify multiple `block_device` entries to create an instance with multiple ephemeral (local) disks. In order to create multiple ephemeral disks, the sum of the total amount of ephemeral space must be less than or equal to what the chosen flavor supports.

The following example shows how to create an instance with multiple ephemeral disks:

```

resource "flexibleengine_compute_instance_v2" "foo" {
  name          = "terraform-test"
  security_groups = ["default"]

  block_device {
    boot_index          = 0
    delete_on_termination = true
    destination_type    = "local"
    source_type         = "image"
    uuid                 = "<image uuid>"
  }

  block_device {
    boot_index          = -1
    delete_on_termination = true
    destination_type    = "local"
    source_type         = "blank"
    volume_size         = 1
  }

  block_device {
    boot_index          = -1
    delete_on_termination = true
    destination_type    = "local"
    source_type         = "blank"
    volume_size         = 1
  }
}

```

Instances and Ports

Neutron Ports are a great feature and provide a lot of functionality. However, there are some notes to be aware of when mixing Instances and Ports:

- When attaching an Instance to one or more networks using Ports, place the security groups on the Port and not the Instance. If you place the security groups on the Instance, the security groups will not be applied upon creation, but they will be applied upon a refresh. This is a known FlexibleEngine bug.
- Network IP information is not available within an instance for networks that are attached with Ports. This is mostly due to the flexibility Neutron Ports provide when it comes to IP addresses. For example, a Neutron Port can have multiple Fixed IP addresses associated with it. It's not possible to know which single IP address the user would want returned to the Instance's state information. Therefore, in order for a Provisioner to connect to an Instance via its network Port, customize the `connection` information:

```
resource "flexibleengine_networking_port_v2" "port_1" {
  name          = "port_1"
  admin_state_up = "true"

  network_id = "0a1d0a27-cffa-4de3-92c5-9d3fd3f2e74d"

  security_group_ids = [
    "2f02d20a-8dca-49b7-b26f-b6ce9fddaf4f",
    "ca1e5ed7-dae8-4605-987b-fadaeeb30461",
  ]
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name = "instance_1"

  network {
    port = "${flexibleengine_networking_port_v2.port_1.id}"
  }

  connection {
    user          = "root"
    host          = "${flexibleengine_networking_port_v2.port_1.fixed_ip.0.ip_address}"
    private_key = "~/path/to/key"
  }

  provisioner "remote-exec" {
    inline = [
      "echo terraform executed > /tmp/foo",
    ]
  }
}
```

flexibleengine_compute_interface_attach_v2

Attaches a Network Interface (a Port) to an Instance using the FlexibleEngine Compute (Nova) v2 API.

Example Usage

Basic Attachment

```
resource "flexibleengine_networking_network_v2" "network_1" {
  name          = "network_1"
  admin_state_up = "true"
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  security_groups = ["default"]
}

resource "flexibleengine_compute_interface_attach_v2" "ai_1" {
  instance_id = "${flexibleengine_compute_instance_v2.instance_1.id}"
  network_id  = "${flexibleengine_networking_port_v2.network_1.id}"
}
```

Attachment Specifying a Fixed IP

```
resource "flexibleengine_networking_network_v2" "network_1" {
  name          = "network_1"
  admin_state_up = "true"
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  security_groups = ["default"]
}

resource "flexibleengine_compute_interface_attach_v2" "ai_1" {
  instance_id = "${flexibleengine_compute_instance_v2.instance_1.id}"
  network_id  = "${flexibleengine_networking_port_v2.network_1.id}"
  fixed_ip    = "10.0.10.10"
}
```

Attachment Using an Existing Port

```

resource "flexibleengine_networking_network_v2" "network_1" {
  name          = "network_1"
  admin_state_up = "true"
}

resource "flexibleengine_networking_port_v2" "port_1" {
  name          = "port_1"
  network_id    = "${flexibleengine_networking_network_v2.network_1.id}"
  admin_state_up = "true"
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  security_groups = ["default"]
}

resource "flexibleengine_compute_interface_attach_v2" "ai_1" {
  instance_id = "${flexibleengine_compute_instance_v2.instance_1.id}"
  port_id     = "${flexibleengine_networking_port_v2.port_1.id}"
}

```

Attaching Multiple Interfaces

```

resource "flexibleengine_networking_network_v2" "network_1" {
  name          = "network_1"
  admin_state_up = "true"
}

resource "flexibleengine_networking_port_v2" "ports" {
  count          = 2
  name          = "${format("port-%02d", count.index + 1)}"
  network_id    = "${flexibleengine_networking_network_v2.network_1.id}"
  admin_state_up = "true"
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  security_groups = ["default"]
}

resource "flexibleengine_compute_interface_attach_v2" "attachments" {
  count          = 2
  instance_id   = "${flexibleengine_compute_instance_v2.instance_1.id}"
  port_id       = "${flexibleengine_networking_port_v2.ports.*.id[count.index]}"
}

```

Note that the above example will not guarantee that the ports are attached in a deterministic manner. The ports will be attached in a seemingly random order.

If you want to ensure that the ports are attached in a given order, create explicit dependencies between the ports, such as:

```

resource "flexibleengine_networking_network_v2" "network_1" {
  name          = "network_1"
  admin_state_up = "true"
}

resource "flexibleengine_networking_port_v2" "ports" {
  count          = 2
  name          = "${format("port-%02d", count.index + 1)}"
  network_id    = "${flexibleengine_networking_network_v2.network_1.id}"
  admin_state_up = "true"
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  security_groups = ["default"]
}

resource "flexibleengine_compute_interface_attach_v2" "ai_1" {
  instance_id = "${flexibleengine_compute_instance_v2.instance_1.id}"
  port_id     = "${flexibleengine_networking_port_v2.ports.*.id[0]}"
}

resource "flexibleengine_compute_interface_attach_v2" "ai_2" {
  instance_id = "${flexibleengine_compute_instance_v2.instance_1.id}"
  port_id     = "${flexibleengine_networking_port_v2.ports.*.id[1]}"
}

```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the interface attachment. If omitted, the `region` argument of the provider is used. Changing this creates a new attachment.
- `instance_id` - (Required) The ID of the Instance to attach the Port or Network to.
- `port_id` - (Optional) The ID of the Port to attach to an Instance. *NOTE:* This option and `network_id` are mutually exclusive.
- `network_id` - (Optional) The ID of the Network to attach to an Instance. A port will be created automatically. *NOTE:* This option and `port_id` are mutually exclusive.
- `fixed_ip` - (Optional) An IP address to associate with the port. *NOTE:* This option cannot be used with `port_id`. You must specify a `network_id`. The IP address must lie in a range on the supplied network.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `instance_id` - See Argument Reference above.

- `port_id` - See Argument Reference above.
- `network_id` - See Argument Reference above.
- `fixed_ip` - See Argument Reference above.

Import

Interface Attachments can be imported using the Instance ID and Port ID separated by a slash, e.g.

```
$ terraform import flexibleengine_compute_interface_attach_v2.ai_1 89c60255-9bd6-460c-822a-e2b959ede9d2/45670584-225f-46c3-b33e-6707b589b666
```

flexibleengine_compute_keypair_v2

Manages a V2 keypair resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_compute_keypair_v2" "test-keypair" {
  name      = "my-keypair"
  public_key = "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDAjpC1hwiOCCmKEWxJ4qzTTsJbKzndLotBCz5PcwtUnflmU+gHJtWMZKpuEGVi29h0A/+ydKek1018k10Ff+4tyFjiHDQAnOfgWf7+b1yK+qDip3X1C0UPMbwHlTfSGWLGZqd9LvEFx9k3h/M+VtMvwR1LJ9LUyTAImnNjWG7TaIPmui30HvM2UifEmqr4ijq45MyX2+fLIePLRIF61p4whjHAQYufqyno3BS48icQb4p6iVEZPo4AE2o9oIyQvj2mx4dk5Y8CgSETOZTYD0R3rU2fZTRDRgPJDH9FwvQjF5tA0p3d9CoWWd2s6GKKbfoUIi8R/Db1BSPJwkqB"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Compute client. Keypairs are associated with accounts, but a Compute client is needed to create one. If omitted, the `region` argument of the provider is used. Changing this creates a new keypair.
- `name` - (Required) A unique name for the keypair. Changing this creates a new keypair.
- `public_key` - (Required) A pregenerated OpenSSH-formatted public key. Changing this creates a new keypair.
- `value_specs` - (Optional) Map of additional options.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `public_key` - See Argument Reference above.

Import

Keypairs can be imported using the `name`, e.g.

```
$ terraform import flexibleengine_compute_keypair_v2.my-keypair test-keypair
```

flexibleengine_compute_servergroup_v2

Manages a V2 Server Group resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_compute_servergroup_v2" "test-sg" {  
  name      = "my-sg"  
  policies = ["anti-affinity"]  
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Compute client. If omitted, the `region` argument of the provider is used. Changing this creates a new server group.
- `name` - (Required) A unique name for the server group. Changing this creates a new server group.
- `policies` - (Required) The set of policies for the server group. Only two two policies are available right now, and both are mutually exclusive. See the Policies section for more information. Changing this creates a new server group.
- `value_specs` - (Optional) Map of additional options.

Policies

- `affinity` - All instances/servers launched in this group will be hosted on the same compute node.
- `anti-affinity` - All instances/servers launched in this group will be hosted on different compute nodes.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `policies` - See Argument Reference above.
- `members` - The instances that are part of this server group.

Import

Server Groups can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_compute_servergroup_v2.test-sg 1bc30ee9-9d5b-4c30-bdd5-7f1e663f5edf
```

flexibleengine_compute_volume_attach_v2

Attaches a Block Storage Volume to an Instance using the FlexibleEngine Compute (Nova) v2 API.

Example Usage

Basic attachment of a single volume to a single instance

```
resource "flexibleengine_blockstorage_volume_v2" "volume_1" {
  name = "volume_1"
  size = 1
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  security_groups = ["default"]
}

resource "flexibleengine_compute_volume_attach_v2" "va_1" {
  instance_id = "${flexibleengine_compute_instance_v2.instance_1.id}"
  volume_id   = "${flexibleengine_blockstorage_volume_v2.volume_1.id}"
}
```

Attaching multiple volumes to a single instance

```
resource "flexibleengine_blockstorage_volume_v2" "volumes" {
  count = 2
  name  = "${format("vol-%02d", count.index + 1)}"
  size  = 1
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  security_groups = ["default"]
}

resource "flexibleengine_compute_volume_attach_v2" "attachments" {
  count          = 2
  instance_id   = "${flexibleengine_compute_instance_v2.instance_1.id}"
  volume_id     = "${element(flexibleengine_blockstorage_volume_v2.volumes.*, count.index)}"
}

output "volume devices" {
  value = "${flexibleengine_compute_volume_attach_v2.attachments.*.device}"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Compute client. A Compute client is needed to create a volume attachment. If omitted, the `region` argument of the provider is used. Changing this creates a new volume attachment.
- `instance_id` - (Required) The ID of the Instance to attach the Volume to.
- `volume_id` - (Required) The ID of the Volume to attach to an Instance.
- `device` - (Optional) The device of the volume attachment (ex: `/dev/vdc`). *NOTE:* Being able to specify a device is dependent upon the hypervisor in use. There is a chance that the device specified in Terraform will not be the same device the hypervisor chose. If this happens, Terraform will wish to update the device upon subsequent applying which will cause the volume to be detached and reattached indefinitely. Please use with caution.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `instance_id` - See Argument Reference above.
- `volume_id` - See Argument Reference above.
- `device` - See Argument Reference above. *NOTE:* The correctness of this information is dependent upon the hypervisor in use. In some cases, this should not be used as an authoritative piece of information.

Import

Volume Attachments can be imported using the Instance ID and Volume ID separated by a slash, e.g.

```
$ terraform import flexibleengine_compute_volume_attach_v2.va_1 89c60255-9bd6-460c-822a-e2b959ede9d2/45670584-225f-46c3-b33e-6707b589b666
```

flexibleengine_csbs_backup_policy_v1

Provides an FlexibleEngine Backup Policy of Resources.

Example Usage

```
variable "name" { }
variable "id" { }
variable "resource_name" { }

resource "flexibleengine_csbs_backup_policy_v1" "backup_policy_v1" {
  name = "${var.name}"
  resource {
    id = "${var.id}"
    type = "OS::Nova::Server"
    name = "${var.resource_name}"
  }
  scheduled_operation {
    enabled = true
    operation_type = "backup"
    trigger_pattern = "BEGIN:VCALENDAR\r\nBEGIN:VEVENT\r\nRRULE:FREQ=WEEKLY;BYDAY=TH;BYHOUR=12;BYMINUTE=
27\r\nEND:VEVENT\r\nEND:VCALENDAR\r\n"
  }
}
```

Argument Reference

The following arguments are supported:

- **name** - (Required) Specifies the name of backup policy. The value consists of 1 to 255 characters and can contain only letters, digits, underscores (_), and hyphens (-).
- **description** - (Optional) Backup policy description. The value consists of 0 to 255 characters and must not contain a greater-than sign (>) or less-than sign (<).
- **provider_id** - (Required) Specifies backup provider ID. Default value is **fc4d5750-22e7-4798-8a46-f48f62c4c1da**
- **common** - (Optional) General backup policy parameters, which are blank by default.
- **scheduled_operation** block supports the following arguments:
 - **name** - (Optional) Specifies Scheduling period name. The value consists of 1 to 255 characters and can contain only letters, digits, underscores (_), and hyphens (-).
 - **description** - (Optional) Specifies Scheduling period description. The value consists of 0 to 255 characters and must not contain a greater-than sign (>) or less-than sign (<).
 - **enabled** - (Optional) Specifies whether the scheduling period is enabled. Default value is **true**
 - **max_backups** - (Optional) Specifies maximum number of backups that can be automatically created for a

backup object.

- `retention_duration_days` - (Optional) Specifies duration of retaining a backup, in days.
- `permanent` - (Optional) Specifies whether backups are permanently retained.
- `trigger_pattern` - (Required) Specifies Scheduling policy of the scheduler.
- `operation_type` - (Required) Specifies Operation type, which can be backup.
- `resource block` supports the following arguments:
 - `id` - (Required) Specifies the ID of the object to be backed up.
 - `type` - (Required) Entity object type of the backup object. If the type is VMs, the value is **OS::Nova::Server**.
 - `name` - (Required) Specifies backup object name.

Attributes Reference

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

- `status` - Status of Backup Policy.
- `id` - Backup Policy ID.
- `scheduled_operation` - Backup plan information
 - `id` - Specifies Scheduling period ID.
 - `trigger_id` - Specifies Scheduler ID.
 - `trigger_name` - Specifies Scheduler name.
 - `trigger_type` - Specifies Scheduler type.

Import

Backup Policy can be imported using `id` , e.g.

```
$ terraform import flexibleengine_csbs_backup_policy_v1.backup_policy_v1 7056d636-ac60-4663-8a6c-82d3c32c1c64
```

flexibleengine_csbs_backup_v1

Provides an FlexibleEngine Backup of Resources.

Example Usage

```
variable "backup_name" { }
variable "resource_id" { }

resource "flexibleengine_csbs_backup_v1" "backup_v1" {
  backup_name = "${var.backup_name}"
  resource_id = "${var.resource_id}"
  resource_type = "OS::Nova::Server"
}
```

Argument Reference

The following arguments are supported:

- `backup_name` - (Optional) Name for the backup. The value consists of 1 to 255 characters and can contain only letters, digits, underscores (`_`), and hyphens (`-`). Changing `backup_name` creates a new backup.
- `description` - (Optional) Backup description. The value consists of 0 to 255 characters and must not contain a greater-than sign (`>`) or less-than sign (`<`). Changing `description` creates a new backup.
- `resource_id` - (Required) ID of the target to which the backup is restored. Changing this creates a new backup.
- `resource_type` - (Optional) Type of the target to which the backup is restored. The default value is `OS::Nova::Server` for an ECS. Changing this creates a new backup.

Attributes Reference

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

- `status` - It specifies the status of backup.
- `backup_record_id` - Specifies backup record ID.
- `volume_backups` block supports the following arguments:
 - `status` - Status of backup Volume.
 - `space_saving_ratio` - Specifies space saving rate.
 - `name` - It gives EVS disk backup name.
 - `bootable` - Specifies whether the disk is bootable.

- `average_speed` - Specifies the average speed.
- `source_volume_size` - Shows source volume size in GB.
- `source_volume_id` - It specifies source volume ID.
- `incremental` - Shows whether incremental backup is used.
- `snapshot_id` - ID of snapshot.
- `source_volume_name` - Specifies source volume name.
- `image_type` - It specifies backup. The default value is backup.
- `id` - Specifies Cinder backup ID.
- `size` - Specifies accumulated size (MB) of backups.
- `vm_metadata` block supports the following arguments:
 - `name` - Name of backup data.
 - `eip` - Specifies elastic IP address of the ECS.
 - `cloud_service_type` - Specifies ECS type.
 - `ram` - Specifies memory size of the ECS, in MB.
 - `vcpus` - Specifies CPU cores corresponding to the ECS.
 - `private_ip` - It specifies internal IP address of the ECS.
 - `disk` - Shows system disk size corresponding to the ECS specifications.
 - `image_type` - Specifies image type.

Import

Backup can be imported using `backup_record_id`, e.g.

```
$ terraform import flexibleengine_csbs_backup_v1.backup_v1.backup_v1 7056d636-ac60-4663-8a6c-82d3c32c1c64
```

flexibleengine_cts_tracker_v1

Allows you to collect, store, and query cloud resource operation records.

Example Usage

```
variable "bucket_name" { }

resource "flexibleengine_cts_tracker_v1" "tracker_v1" {
  bucket_name      = "${var.bucket_name}"
  file_prefix_name = "y08Q"
}
```

Argument Reference

The following arguments are supported:

- `bucket_name` - (Required) The OBS bucket name for a tracker.
- `file_prefix_name` - (Optional) The prefix of a log that needs to be stored in an OBS bucket.
- `status` - The status of a tracker. The value should be **enabled** when creating a tracker, and when updating the value can be enabled or disabled.

Attributes Reference

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

- `tracker_name` - The tracker name. Currently, only tracker **system** is available.

Import

CTS tracker can be imported using `tracker_name`, e.g.

```
$ terraform import flexibleengine_cts_tracker_v1.tracker system
```

flexibleengine_dcs_instance_v1

Manages a DCS instance in the flexibleengine DCS Service.

Example Usage

Automatically detect the correct network

```
resource "flexibleengine_networking_secgroup_v2" "secgroup_1" {
  name = "secgroup_1"
  description = "secgroup_1"
}
data "flexibleengine_dcs_az_v1" "az_1" {
  port = "8002"
}
data "flexibleengine_dcs_product_v1" "product_1" {
  spec_code = "dcs.master_standby"
}
resource "flexibleengine_dcs_instance_v1" "instance_1" {
  name = "test_dcs_instance"
  engine_version = "3.0.7"
  password = "Huawei_test"
  engine = "Redis"
  capacity = 2
  vpc_id = "1477393a-29c9-4de5-843f-18ef51257c7e"
  security_group_id = "${flexibleengine_networking_secgroup_v2.secgroup_1.id}"
  subnet_id = "27d99e17-42f2-4751-818f-5c8c6c03ff15"
  available_zones = ["${data.flexibleengine_dcs_az_v1.az_1.id}"]
  product_id = "${data.flexibleengine_dcs_product_v1.product_1.id}"
  save_days = 1
  backup_type = "manual"
  begin_at = "00:00-01:00"
  period_type = "weekly"
  backup_at = [1]
  depends_on = ["data.flexibleengine_dcs_product_v1.product_1", "flexibleengine_networking_secgroup_v2.secgroup_1"]
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) Indicates the name of an instance. An instance name starts with a letter, consists of 4 to 64 characters, and supports only letters, digits, and hyphens (-).
- `description` - (Optional) Indicates the description of an instance. It is a character string containing not more than 1024 characters.
- `engine` - (Required) Indicates a cache engine. Only Redis is supported. Changing this creates a new instance.

- `engine_version` - (Required) Indicates the version of a cache engine, which is 3.0.7. Changing this creates a new instance.
- `capacity` - (Required) Indicates the Cache capacity. Unit: GB. For a DCS Redis or Memcached instance in single-node or master/standby mode, the cache capacity can be 2 GB, 4 GB, 8 GB, 16 GB, 32 GB, or 64 GB. For a DCS Redis instance in cluster mode, the cache capacity can be 64, 128, 256, 512, or 1024 GB. Changing this creates a new instance.
- `access_user` - (Optional) Username used for accessing a DCS instance after password authentication. A username starts with a letter, consists of 1 to 64 characters, and supports only letters, digits, and hyphens (-). Changing this creates a new instance.
- `password` - (Required) Password of a DCS instance. The password of a DCS Redis instance must meet the following complexity requirements: Changing this creates a new instance.
- `vpc_id` - (Required) Tenant's VPC ID. For details on how to create VPCs, see the Virtual Private Cloud API Reference. Changing this creates a new instance.
- `security_group_id` - (Required) Tenant's security group ID. For details on how to create security groups, see the Virtual Private Cloud API Reference.
- `subnet_id` - (Required) Subnet ID. For details on how to create subnets, see the Virtual Private Cloud API Reference. Changing this creates a new instance.
- `available_zones` - (Required) IDs of the AZs where cache nodes reside. For details on how to query AZs, see Querying AZ Information. Changing this creates a new instance.
- `product_id` - (Required) Product ID used to differentiate DCS instance types. Changing this creates a new instance.
- `maintain_begin` - (Optional) Indicates the time at which a maintenance time window starts. Format: HH:mm:ss. The start time and end time of a maintenance time window must indicate the time segment of a supported maintenance time window. For details, see section Querying Maintenance Time Windows. The start time must be set to 22:00, 02:00, 06:00, 10:00, 14:00, or 18:00. Parameters `maintain_begin` and `maintain_end` must be set in pairs. If parameter `maintain_begin` is left blank, parameter `maintain_end` is also blank. In this case, the system automatically allocates the default start time 02:00.
- `maintain_end` - (Optional) Indicates the time at which a maintenance time window ends. Format: HH:mm:ss. The start time and end time of a maintenance time window must indicate the time segment of a supported maintenance time window. For details, see section Querying Maintenance Time Windows. The end time is four hours later than the start time. For example, if the start time is 22:00, the end time is 02:00. Parameters `maintain_begin` and `maintain_end` must be set in pairs. If parameter `maintain_end` is left blank, parameter `maintain_begin` is also blank. In this case, the system automatically allocates the default end time 06:00.
- `save_days` - (Optional) Retention time. Unit: day. Range: 1–7. Changing this creates a new instance.
- `backup_type` - (Optional) Backup type. Options: auto: automatic backup. manual: manual backup. Changing this creates a new instance.
- `begin_at` - (Optional) Time at which backup starts. "00:00-01:00" indicates that backup starts at 00:00:00. Changing this creates a new instance.
- `period_type` - (Optional) Interval at which backup is performed. Currently, only weekly backup is supported. Changing this creates a new instance.
- `backup_at` - (Optional) Day in a week on which backup starts. Range: 1–7. Where: 1 indicates Monday; 7 indicates Sunday. Changing this creates a new instance.

Attributes Reference

The following attributes are exported:

- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `engine` - See Argument Reference above.
- `engine_version` - See Argument Reference above.
- `capacity` - See Argument Reference above.
- `access_user` - See Argument Reference above.
- `password` - See Argument Reference above.
- `vpc_id` - See Argument Reference above.
- `vpc_name` - Indicates the name of a vpc.
- `security_group_id` - See Argument Reference above.
- `security_group_name` - Indicates the name of a security group.
- `subnet_id` - See Argument Reference above.
- `subnet_name` - Indicates the name of a subnet.
- `available_zones` - See Argument Reference above.
- `product_id` - See Argument Reference above.
- `maintain_begin` - See Argument Reference above.
- `maintain_end` - See Argument Reference above.
- `save_days` - See Argument Reference above.
- `backup_type` - See Argument Reference above.
- `begin_at` - See Argument Reference above.
- `period_type` - See Argument Reference above.
- `backup_at` - See Argument Reference above.
- `order_id` - An order ID is generated only in the monthly or yearly billing mode. In other billing modes, no value is returned for this parameter.
- `port` - Port of the cache node.
- `resource_spec_code` - Resource specifications. `dc.single_node`: indicates a DCS instance in single-node mode. `dc.master_standby`: indicates a DCS instance in master/standby mode. `dc.cluster`: indicates a DCS instance in cluster mode.
- `used_memory` - Size of the used memory. Unit: MB.
- `internal_version` - Internal DCS version.

- `max_memory` - Overall memory size. Unit: MB.
- `user_id` - Indicates a user ID.
- `ip` - Cache node's IP address in tenant's VPC.

flexibleengine_dds_instance_v3

Manages dds instance resource within FlexibleEngine

Example Usage: Creating a Cluster Community Edition

```
resource "flexibleengine_dds_instance_v3" "instance" {
  name = "dds-instance"
  datastore {
    type = "DDS-Community"
    version = "3.4"
    storage_engine = "wiredTiger"
  }
  region = "eu-west-0"
  availability_zone = "eu-west-0a"
  vpc_id = "c1095fe7-03df-4205-ad2d-6f4c181d436e"
  subnet_id = "b65f8d25-c533-47e2-8601-cfaa265a3e3e"
  security_group_id = "e28c7982-ecf0-4498-852d-9683cfc364f2"
  password = "Test@123"
  mode = "Sharding"
  flavor {
    type = "mongos"
    num = 2
    spec_code = "dds.mongodb.s3.medium.4.mongos"
  }
  flavor {
    type = "shard"
    num = 2
    storage = "ULTRAHIGH"
    size = 20
    spec_code = "dds.mongodb.s3.medium.4.shard"
  }
  flavor {
    type = "config"
    num = 1
    storage = "ULTRAHIGH"
    size = 20
    spec_code = "dds.mongodb.s3.large.2.config"
  }
  backup_strategy {
    start_time = "08:00-09:00"
    keep_days = "8"
  }
}
```

Example Usage: Creating a Replica Set

```

resource "flexibleengine_dds_instance_v3" "instance" {
  name = "dds-instance"
  datastore {
    type = "DDS-Community"
    version = "3.4"
    storage_engine = "wiredTiger"
  }
  region = "eu-west-0"
  availability_zone = "eu-west-0a"
  vpc_id = "c1095fe7-03df-4205-ad2d-6f4c181d436e"
  subnet_id = "b65f8d25-c533-47e2-8601-cfaa265a3e3e"
  security_group_id = "e28c7982-ecf0-4498-852d-9683cfc364f2"
  password = "Test@123"
  mode = "ReplicaSet"
  flavor {
    type = "replica"
    num = 1
    storage = "ULTRAHIGH"
    size = 30
    spec_code = "dds.mongodb.s3.medium.4.repset"
  }
  backup_strategy {
    start_time = "08:00-09:00"
    keep_days = "8"
  }
}

```

Argument Reference

The following arguments are supported:

- `region` - (Optional) Specifies the region of the DDS instance. Changing this creates a new instance.
- `name` - (Required) Specifies the DB instance name. The DB instance name of the same type is unique in the same tenant. Changing this creates a new instance.
- `datastore` - (Required) Specifies database information. The structure is described below. Changing this creates a new instance.
- `availability_zone` - (Required) Specifies the ID of the availability zone. Changing this creates a new instance.
- `vpc_id` - (Required) Specifies the VPC ID. For details about how to obtain this parameter value, see section "Virtual Private Cloud" in the Virtual Private Cloud API Reference. Changing this creates a new instance.
- `subnet_id` - (Required) Specifies the subnet Network ID. Changing this creates a new instance.
- `security_group_id` - (Required) Specifies the security group ID of the DDS instance. Changing this creates a new instance.
- `password` - (Required) Specifies the Administrator password of the database instance. Changing this creates a new instance.
- `disk_encryption_id` - (Required) Specifies the disk encryption ID of the instance. Changing this creates a new instance.

- `mode` - (Required) Specifies the mode of the database instance. Changing this creates a new instance.
- `flavor` - (Required) Specifies the flavors information. The structure is described below. Changing this creates a new instance.
- `backup_strategy` - (Optional) Specifies the advanced backup policy. The structure is described below. Changing this creates a new instance.

The `datastore` block supports:

- `type` - (Required) Specifies the DB engine. Only DDS-Community is supported now.
- `version` - (Required) Specifies the DB instance version. Only 3.4 is supported now.
- `storage_engine` - (Optional) Specifies the storage engine of the DB instance. Only wiredTiger is supported now.

The `flavor` block supports:

- `type` - (Required) Specifies the node type. Valid value: mongos, shard, config, replica.
- `num` - (Required) Specifies the node quantity. Valid value:
 - the number of mongos ranges from 2 to 12.
 - the number of shard ranges from 2 to 12.
 - config: the value is 1.
 - replica: the value is 1.
- `storage` - (Optional) Specifies the disk type. Valid value: ULTRAHIGH which indicates the type SSD.
- `size` - (Optional) Specifies the disk size. The value must be a multiple of 10. The unit is GB. This parameter is mandatory for nodes except mongos and invalid for mongos.
- `spec_code` - (Required) Specifies the resource specification code. Valid values:

engine_name | type | vcpus | ram | speccode ---- | --- | --- DDS-Community | mongos | 1 | 4 |
 dds.mongodb.s3.medium.4.mongos DDS-Community | mongos | 2 | 8 | dds.mongodb.s3.large.4.mongos DDS-Community
 | mongos | 4 | 16 | dds.mongodb.s3.xlarge.4.mongos DDS-Community | mongos | 8 | 32 |
 dds.mongodb.s3.2xlarge.4.mongos DDS-Community | mongos | 16 | 64 | dds.mongodb.s3.4xlarge.4.mongos DDS-
 Community | shard | 1 | 4 | dds.mongodb.s3.medium.4.shard DDS-Community | shard | 2 | 8 |
 dds.mongodb.s3.large.4.shard DDS-Community | shard | 4 | 16 | dds.mongodb.s3.xlarge.4.shard DDS-Community | shard
 | 8 | 32 | dds.mongodb.s3.2xlarge.4.shard DDS-Community | shard | 16 | 64 | dds.mongodb.s3.4xlarge.4.shard DDS-
 Community | config | 2 | 4 | dds.mongodb.s3.large.2.config DDS-Community | replica | 1 | 4 |
 dds.mongodb.s3.medium.4.repset DDS-Community | replica | 2 | 8 | dds.mongodb.s3.large.4.repset DDS-Community |
 replica | 4 | 16 | dds.mongodb.s3.xlarge.4.repset DDS-Community | replica | 8 | 32 | dds.mongodb.s3.2xlarge.4.repset
 DDS-Community | replica | 16 | 64 | dds.mongodb.s3.4xlarge.4.repset

The `backup_strategy` block supports:

- `start_time` - (Required) Specifies the backup time window. Automated backups will be triggered during the backup time window. The value cannot be empty. It must be a valid value in the "hh:mm-HH:MM" format. The current time is in the UTC format.
 - The HH value must be 1 greater than the hh value.

- The values from mm and MM must be the same and must be set to any of the following 00, 15, 30, or 45.
- keep_days - (Optional) Specifies the number of days to retain the generated backup files. The value range is from 0 to 732.
 - If this parameter is set to 0, the automated backup policy is not set.
 - If this parameter is not transferred, the automated backup policy is enabled by default. Backup files are stored for seven days by default.

Attributes Reference

The following attributes are exported:

- region - See Argument Reference above.
- name - See Argument Reference above.
- datastore - See Argument Reference above.
- availability_zone - See Argument Reference above.
- vpc_id - See Argument Reference above.
- subnet_id - See Argument Reference above.
- security_group_id - See Argument Reference above.
- password - See Argument Reference above.
- disk_encryption_id - See Argument Reference above.
- mode - See Argument Reference above.
- flavor - See Argument Reference above.
- backup_strategy - See Argument Reference above.
- db_username - Indicates the DB Administrator name.

flexibleengine_dns_recordset_v2

Manages a DNS record set in the FlexibleEngine DNS Service.

Example Usage

Automatically detect the correct network

```
resource "flexibleengine_dns_zone_v2" "example_zone" {
  name = "example.com."
  email = "email2@example.com"
  description = "a zone"
  ttl = 6000
  zone_type = "public"
}

resource "flexibleengine_dns_recordset_v2" "rs_example_com" {
  zone_id = "${flexibleengine_dns_zone_v2.example_zone.id}"
  name = "rs.example.com."
  description = "An example record set"
  ttl = 3000
  type = "A"
  records = ["10.0.0.1"]
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 DNS client. If omitted, the `region` argument of the provider is used. Changing this creates a new DNS record set.
- `zone_id` - (Required) The ID of the zone in which to create the record set. Changing this creates a new DNS record set.
- `name` - (Required) The name of the record set. Note the `.` at the end of the name. Changing this creates a new DNS record set.
- `type` - (Optional) The type of record set. The options include `A`, `AAAA`, `MX`, `CNAME`, `TXT`, `NS`, `SRV`, and `PTR`. Changing this creates a new DNS record set.
- `ttl` - (Optional) The time to live (TTL) of the record set (in seconds). The value range is 300–2147483647. The default value is 300.
- `description` - (Optional) A description of the record set.
- `records` - (Required) An array of DNS records.
- `value_specs` - (Optional) Map of additional options. Changing this creates a new record set.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `type` - See Argument Reference above.
- `ttl` - See Argument Reference above.
- `description` - See Argument Reference above.
- `records` - See Argument Reference above.
- `zone_id` - See Argument Reference above.
- `value_specs` - See Argument Reference above.

Import

This resource can be imported by specifying the zone ID and recordset ID, separated by a forward slash.

```
$ terraform import flexibleengine_dns_recordset_v2.recordset_1 <zone_id>/<recordset_id>
```

flexibleengine_dns_zone_v2

Manages a DNS zone in the FlexibleEngine DNS Service.

Example Usage

Create a public DNS zone

```
resource "flexibleengine_dns_zone_v2" "my_public_zone" {
  name = "example.com."
  email = "jdoe@example.com"
  description = "An example zone"
  ttl = 3000
  zone_type = "public"
}
```

Create a private DNS zone

```
resource "flexibleengine_dns_zone_v2" "my_private_zone" {
  name = "1.example.com."
  email = "jdoe@example.com"
  description = "An example zone"
  ttl = 3000
  zone_type = "private"
  router = [
    {router_region = "cn-north-1"
      router_id = "2c1fe4bd-ebad-44ca-ae9d-e94e63847b75"}]
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Compute client. Keypairs are associated with accounts, but a Compute client is needed to create one. If omitted, the `region` argument of the provider is used. Changing this creates a new DNS zone. Changing this creates a new DNS zone.
- `name` - (Required) The name of the zone. Note the `.` at the end of the name. Changing this creates a new DNS zone.
- `email` - (Optional) The email contact for the zone record.
- `zone_type` - (Optional) The type of zone. Can either be `public` or `private`. Changing this creates a new DNS zone.
- `router` - (Optional) Router configuration block which is required if `zone_type` is `private`. The router structure is documented below.

- `ttl` - (Optional) The time to live (TTL) of the zone.
- `description` - (Optional) A description of the zone.
- `value_specs` - (Optional) Map of additional options. Changing this creates a new DNS zone.

The `router` block supports:

- `router_id` - (Required) The router UUID.
- `router_region` - (Required) The region of the router.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `email` - See Argument Reference above.
- `zone_type` - See Argument Reference above.
- `ttl` - See Argument Reference above.
- `description` - See Argument Reference above.
- `masters` - An array of master DNS servers.
- `value_specs` - See Argument Reference above.

Import

This resource can be imported by specifying the zone ID:

```
$ terraform import flexibleengine_dns_zone_v2.zone_1 <zone_id>
```

flexibleengine_drs_replicationconsistencygroup_v2

Manages a V2 replicationconsistencygroup resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_blockstorage_volume_v2" "volume_1" {
  name = "volume_1"
  size = 1
  availability_zone = "eu-west-0a"
}

resource "flexibleengine_blockstorage_volume_v2" "volume_2" {
  name = "volume_2"
  size = 1
  availability_zone = "eu-west-0b"
}

resource "flexibleengine_drs_replication_v2" "replication_1" {
  name = "replication_1"
  description = "The description of replication_1"
  volume_ids = ["${flexibleengine_blockstorage_volume_v2.volume_1.id}", "${flexibleengine_blockstorage_volume_v2.volume_2.id}"]
  priority_station = "eu-west-0a"
}

resource "flexibleengine_drs_replicationconsistencygroup_v2" "replicationconsistencygroup_1" {
  name = "replicationconsistencygroup_1"
  description = "The description of replicationconsistencygroup_1"
  replication_ids = ["${flexibleengine_drs_replication_v2.replication_1.id}"]
  priority_station = "eu-west-0a"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Optional) The name of the replication consistency group. The name can contain a maximum of 255 bytes.
- `description` - (Optional) The description of the replication consistency group. The description can contain a maximum of 255 bytes.
- `replication_ids` - (Required) An array of one or more IDs of the EVS replication pairs used to create the replication consistency group.
- `priority_station` - (Required) The primary AZ of the replication consistency group. That is the AZ where the production disk belongs.
- `replication_model` - (Optional) The type of the created replication consistency group. Currently only type hypermetro is supported.

Attributes Reference

The following attributes are exported:

- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `replication_ids` - See Argument Reference above.
- `priority_station` - See Argument Reference above.
- `replication_model` - See Argument Reference above.
- `status` - The status of the replication consistency group.
- `replication_status` - The replication status of the replication consistency group.
- `created_at` - The creation time of the replication consistency group.
- `updated_at` - The update time of the replication consistency group.
- `failure_detail` - The returned error code if the replication consistency group status is error.
- `fault_level` - The fault level of the replication consistency group.

flexibleengine_drs_replication_v2

Manages a V2 replication resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_blockstorage_volume_v2" "volume_1" {
  name = "volume_1"
  size = 1
  availability_zone = "eu-west-0a"
}

resource "flexibleengine_blockstorage_volume_v2" "volume_2" {
  name = "volume_2"
  size = 1
  availability_zone = "eu-west-0b"
}

resource "flexibleengine_drs_replication_v2" "replication_1" {
  name = "replication_1"
  description = "The description of replication_1"
  volume_ids = ["${flexibleengine_blockstorage_volume_v2.volume_1.id}", "${flexibleengine_blockstorage_volume_v2.volume_2.id}"]
  priority_station = "eu-west-0a"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Optional) The name of the EVS replication pair. The name can contain a maximum of 255 bytes.
- `description` - (Optional) The description of the EVS replication pair. The description can contain a maximum of 255 bytes.
- `volume_ids` - (Required) An array of one or more IDs of the EVS disks used to create the EVS replication pair.
- `priority_station` - (Required) The primary AZ of the EVS replication pair. That is the AZ where the production disk belongs.
- `replication_model` - (Optional) The type of the EVS replication pair. Currently only type hypermetro is supported.

Attributes Reference

The following attributes are exported:

- `name` - See Argument Reference above.
- `description` - See Argument Reference above.

- `volume_ids` - See Argument Reference above.
- `priority_station` - See Argument Reference above.
- `replication_model` - See Argument Reference above.
- `status` - The status of the EVS replication pair.
- `replication_consistency_group_id` - The ID of the replication consistency group where the EVS replication pair belongs.
- `created_at` - The creation time of the EVS replication pair.
- `updated_at` - The update time of the EVS replication pair.
- `replication_status` - The replication status of the EVS replication pair.
- `progress` - The synchronization progress of the EVS replication pair. Unit: %.
- `failure_detail` - The returned error code if the EVS replication pair status is error.
- `record_metadata` - The metadata of the EVS replication pair.
- `fault_level` - The fault level of the EVS replication pair.

flexibleengine_dws_cluster_v1

Manages a DWS cluster resource within FlexibleEngine

Example Usage

```
resource "flexibleengine_dws_cluster_v1" "cluster" {
  node_type = "dws.d1.xlarge"
  number_of_node = 3
  subnet_id = "{{ subnet_id }}"
  vpc_id = "{{ vpc_id }}"
  security_group_id = "{{ security_group_id }}"
  availability_zone = "{{ availability_zone }}"
  name = "terraform_dws_cluster_test"
  user_name = "test_cluster_admin"
  user_pwd = "cluster123@!"

  timeouts {
    create = "30m"
    delete = "30m"
  }
}
```

Argument Reference

The following arguments are supported:

- `availability_zone` - (Optional) AZ in a cluster
- `name` - (Required) Cluster name, which must be unique and contains 4 to 64 characters, which consist of letters, digits, hyphens (-), or underscores (_) only and must start with a letter.
- `node_type` - (Required) Node type.
- `number_of_node` - (Required) Number of nodes in a cluster. The value ranges from 3 to 32.
- `port` - (Optional) Service port of a cluster (8000 to 10000). The default value is 8000.
- `public_ip` - (Optional) Public IP address. If the value is not specified, public connection is not used by default.
- `security_group_id` - (Required) ID of a security group. The ID is used for configuring cluster network.
- `subnet_id` - (Required) Subnet ID, which is used for configuring cluster network.
- `user_name` - (Required) Administrator username for logging in to a data warehouse cluster The administrator username must:

Consist of lowercase letters, digits, or underscores.

Start with a lowercase letter or an underscore.

Contain 1 to 63 characters.

Cannot be a keyword of the DWS database.

- `user_pwd` - (Required) Administrator password for logging in to a data warehouse cluster

A password must conform to the following rules:

Contains 8 to 32 characters.

Cannot be the same as the username or the username written in reverse order.

Contains three types of the following:

Lowercase letters

Uppercase letters

Digits

Special characters `~!@#%&*()-_+=+|[];:,<.>/?`

- `vpc_id` - (Required) VPC ID, which is used for configuring cluster network.

The `public_ip` block supports:

- `eip_id` - (Optional) EIP ID
- `public_bind_type` - (Optional) Binding type of an EIP. The value can be either of the following:

`auto_assign`

`not_use`

`bind_existing`

The default value is `not_use`.

Attributes Reference

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

- `name` - See Argument Reference above.
- `number_of_node` - See Argument Reference above.
- `availability_zone` - See Argument Reference above.
- `subnet_id` - See Argument Reference above.
- `user_name` - See Argument Reference above.
- `security_group_id` - See Argument Reference above.
- `public_ip` - See Argument Reference above.
- `node_type` - See Argument Reference above.
- `vpc_id` - See Argument Reference above.
- `port` - See Argument Reference above.

- `created` - Cluster creation time. The format is ISO8601:YYYY-MM-DDThh:mm:ssZ.
- `endpoints` - View the private network connection information about the cluster.
- `id` - Cluster ID
- `public_endpoints` - Public network connection information about the cluster. If the value is not specified, the public network connection information is not used by default.
- `status` - Cluster status, which can be one of the following:

CREATING

AVAILABLE

UNAVAILABLE

CREATION FAILED

- `sub_status` - Sub-status of clusters in the AVAILABLE state. The value can be one of the following:

NORMAL

READONLY

REDISTRIBUTING

REDISTRIBUTION-FAILURE

UNBALANCED

UNBALANCED | READONLY

DEGRADED

DEGRADED | READONLY

DEGRADED | UNBALANCED

UNBALANCED | REDISTRIBUTING

UNBALANCED | REDISTRIBUTION-FAILURE

READONLY | REDISTRIBUTION-FAILURE

UNBALANCED | READONLY | REDISTRIBUTION-FAILURE

DEGRADED | REDISTRIBUTION-FAILURE

DEGRADED | UNBALANCED | REDISTRIBUTION-FAILURE

DEGRADED | UNBALANCED | READONLY | REDISTRIBUTION-FAILURE

DEGRADED | UNBALANCED | READONLY

- `task_status` - Cluster management task. The value can be one of the following:

RESTORING

SNAPSHOTTING

GROWING

REBOOTING

SETTING_CONFIGURATION

CONFIGURING_EXT_DATASOURCE

DELETING_EXT_DATASOURCE

REBOOT_FAILURE

RESIZE_FAILURE

- `updated` - Last modification time of a cluster. The format is ISO8601:YYYY-MM-DDThh:mm:ssZ.
- `version` - Data warehouse version

The `endpoints` block supports:

- `connect_info` - Private network connection information
- `jdbc_url` - JDBC URL. The following is the default format:

```
jdbc:postgresql://< connect_info>/
```

The `public_endpoints` block supports:

- `public_connect_info` - Public network connection information
- `jdbc_url` - JDBC URL. The following is the default format:

```
jdbc:postgresql://< public_connect_info>/
```

flexibleengine_elb_backend

Manages an elastic loadbalancer backend resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_elb_loadbalancer" "elb" {
  name = "elb"
  type = "External"
  description = "test elb"
  vpc_id = "e346dc4a-d9a6-46f4-90df-10153626076e"
  admin_state_up = 1
  bandwidth = 5
}

resource "flexibleengine_elb_listener" "listener" {
  name = "test-elb-listener"
  description = "great listener"
  protocol = "TCP"
  backend_protocol = "TCP"
  protocol_port = 12345
  backend_port = 8080
  lb_algorithm = "roundrobin"
  loadbalancer_id = "${flexibleengine_elb_loadbalancer.elb.id}"
  timeouts {
    create = "5m"
    update = "5m"
    delete = "5m"
  }
}

resource "flexibleengine_elb_backend" "backend" {
  address = "192.168.0.211"
  listener_id = "${flexibleengine_elb_listener.listener.id}"
  server_id = "8f7a32f1-f66c-4d13-9b17-3a13f9f0bb8d"
}
```

Argument Reference

The following arguments are supported:

- `listener_id` - (Required) Specifies the listener ID.
- `server_id` - (Required) Specifies the backend member ID.
- `address` - (Required) Specifies the private IP address of the backend member.

Attributes Reference

The following attributes are exported:

- `listener_id` - See Argument Reference above.
- `server_id` - See Argument Reference above.
- `address` - See Argument Reference above.
- `server_address` - Specifies the floating IP address assigned to the backend member.
- `id` - Specifies the backend member ID.
- `status` - Specifies the backend ECS status. The value is ACTIVE, PENDING, or ERROR.
- `health_status` - Specifies the health check status. The value is NORMAL, ABNORMAL, or UNAVAILABLE.
- `update_time` - Specifies the time when information about the backend member was updated.
- `create_time` - Specifies the time when the backend member was created.
- `server_name` - Specifies the backend member name.
- `listeners` - Specifies the listener to which the backend member belongs.

flexibleengine_elb_health

Manages an elastic loadbalancer health resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_elb_loadbalancer" "elb" {
  name = "elb"
  type = "External"
  description = "test elb"
  vpc_id = "e346dc4a-d9a6-46f4-90df-10153626076e"
  admin_state_up = 1
  bandwidth = 5
}

resource "flexibleengine_elb_listener" "listener" {
  name = "test-elb-listener"
  description = "great listener"
  protocol = "TCP"
  backend_protocol = "TCP"
  protocol_port = 12345
  backend_port = 8080
  lb_algorithm = "roundrobin"
  loadbalancer_id = "${flexibleengine_elb_loadbalancer.elb.id}"
  timeouts {
    create = "5m"
    update = "5m"
    delete = "5m"
  }
}

resource "flexibleengine_elb_health" "healthcheck" {
  listener_id = "${flexibleengine_elb_listener.listener.id}"
  healthcheck_protocol = "TCP"
  healthcheck_connect_port = 22
  healthy_threshold = 5
  healthcheck_timeout = 25
  healthcheck_interval = 3
  timeouts {
    create = "5m"
    update = "5m"
    delete = "5m"
  }
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the elb health. If omitted, the `region` argument of the provider is used. Changing this creates a new elb health.

- `listener_id` - (Required) Specifies the ID of the listener to which the health check task belongs.
- `healthcheck_protocol` - (Optional) Specifies the protocol used for the health check. The value can be HTTP or TCP (case-insensitive).
- `healthcheck_uri` - (Optional) Specifies the URI for health check. This parameter is valid when `healthcheck_protocol` is HTTP. The value is a string of 1 to 80 characters that must start with a slash (/) and can only contain letters, digits, and special characters, such as `-.%?#&`.
- `healthcheck_connect_port` - (Optional) Specifies the port used for the health check. The value ranges from 1 to 65535.
- `healthy_threshold` - (Optional) Specifies the threshold at which the health check result is success, that is, the number of consecutive successful health checks when the health check result of the backend server changes from fail to success. The value ranges from 1 to 10.
- `unhealthy_threshold` - (Optional) Specifies the threshold at which the health check result is fail, that is, the number of consecutive failed health checks when the health check result of the backend server changes from success to fail. The value ranges from 1 to 10.
- `healthcheck_timeout` - (Optional) Specifies the maximum timeout duration (s) for the health check. The value ranges from 1 to 50.
- `healthcheck_interval` - (Optional) Specifies the maximum interval (s) for health check. The value ranges from 1 to 5.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `listener_id` - See Argument Reference above.
- `healthcheck_protocol` - See Argument Reference above.
- `healthcheck_uri` - See Argument Reference above.
- `healthcheck_connect_port` - See Argument Reference above.
- `healthy_threshold` - See Argument Reference above.
- `unhealthy_threshold` - See Argument Reference above.
- `healthcheck_timeout` - See Argument Reference above.
- `healthcheck_interval` - See Argument Reference above.
- `id` - Specifies the health check task ID.

flexibleengine_elb_listener

Manages an elastic loadbalancer listener resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_elb_loadbalancer" "elb" {
  name = "elb"
  type = "External"
  description = "test elb"
  vpc_id = "e346dc4a-d9a6-46f4-90df-10153626076e"
  admin_state_up = 1
  bandwidth = 5
}

resource "flexibleengine_elb_listener" "listener" {
  name = "test-elb-listener"
  description = "great listener"
  protocol = "TCP"
  backend_protocol = "TCP"
  protocol_port = 12345
  backend_port = 8080
  lb_algorithm = "roundrobin"
  loadbalancer_id = "${flexibleengine_elb_loadbalancer.elb.id}"
  timeouts {
    create = "5m"
    update = "5m"
    delete = "5m"
  }
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the elb listener. If omitted, the `region` argument of the provider is used. Changing this creates a new elb listener.
- `name` - (Required) Specifies the load balancer name. The name is a string of 1 to 64 characters that consist of letters, digits, underscores (`_`), and hyphens (`-`).
- `description` - (Optional) Provides supplementary information about the listener. The value is a string of 0 to 128 characters and cannot be `<>`.
- `loadbalancer_id` - (Required) Specifies the ID of the load balancer to which the listener belongs.
- `protocol` - (Required) Specifies the listening protocol used for layer 4 or 7. The value can be HTTP, TCP, HTTPS, or UDP.
- `protocol_port` - (Required) Specifies the listening port. The value ranges from 1 to 65535.

- `backend_protocol` - (Required) Specifies the backend protocol. If the value of `protocol` is UDP, the value of this parameter can only be UDP. The value can be HTTP, TCP, or UDP.
- `backend_port` - (Required) Specifies the backend port. The value ranges from 1 to 65535.
- `lb_algorithm` - (Required) Specifies the load balancing algorithm for the listener. The value can be `roundrobin`, `leastconn`, or `source`.
- `session_sticky` - (Optional) Specifies whether to enable sticky session. The value can be `true` or `false`. The Sticky session is enabled when the value is `true`, and is disabled when the value is `false`. If the value of `protocol` is HTTP, HTTPS, or TCP, and the value of `lb_algorithm` is not `roundrobin`, the value of this parameter can only be `false`.
- `session_sticky_type` - (Optional) Specifies the cookie processing method. The value is `insert`. `insert` indicates that the cookie is inserted by the load balancer. This parameter is valid when `protocol` is set to HTTP, and `session_sticky` to `true`. The default value is `insert`. This parameter is invalid when `protocol` is set to TCP or UDP, which means the parameter is empty.
- `cookie_timeout` - (Optional) Specifies the cookie timeout period (minutes). This parameter is valid when `protocol` is set to HTTP, `session_sticky` to `true`, and `session_sticky_type` to `insert`. This parameter is invalid when `protocol` is set to TCP or UDP. The value ranges from 1 to 1440.
- `tcp_timeout` - (Optional) Specifies the TCP timeout period (minutes). This parameter is valid when `protocol` is set to TCP. The value ranges from 1 to 5.
- `tcp_draining` - (Optional) Specifies whether to maintain the TCP connection to the backend ECS after the ECS is deleted. This parameter is valid when `protocol` is set to TCP. The value can be `true` or `false`.
- `tcp_draining_timeout` - (Optional) Specifies the timeout duration (minutes) for the TCP connection to the backend ECS after the ECS is deleted. This parameter is valid when `protocol` is set to TCP, and `tcp_draining` to `true`. The value ranges from 0 to 60.
- `certificate_id` - (Optional) Specifies the ID of the SSL certificate used for security authentication when HTTPS is used to make API calls. This parameter is mandatory if the value of `protocol` is HTTPS. The value can be obtained by viewing the details of the SSL certificate.
- `udp_timeout` - (Optional) Specifies the UDP timeout duration (minutes). This parameter is valid when `protocol` is set to UDP. The value ranges from 1 to 1440.
- `ssl_protocols` - (Optional) Specifies the SSL protocol standard supported by a tracker, which is used for enabling specified encryption protocols. This parameter is valid only when the value of `protocol` is set to HTTPS. The value is `TLSv1.2` or `TLSv1.2 TLSv1.1 TLSv1`. The default value is `TLSv1.2`.
- `ssl_ciphers` - (Optional) Specifies the cipher suite of an encryption protocol. This parameter is valid only when the value of `protocol` is set to HTTPS. The value is `Default`, `Extended`, or `Strict`. The default value is `Default`. The value can only be set to `Extended` if the value of `ssl_protocols` is set to `TLSv1.2 TLSv1.1 TLSv1`.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.

- `description` - See Argument Reference above.
- `loadbalancer_id` - See Argument Reference above.
- `protocol` - See Argument Reference above.
- `protocol_port` - See Argument Reference above.
- `backend_protocol` - See Argument Reference above.
- `backend_port` - See Argument Reference above.
- `lb_algorithm` - See Argument Reference above.
- `session_sticky` - See Argument Reference above.
- `session_sticky_type` - See Argument Reference above.
- `cookie_timeout` - See Argument Reference above.
- `tcp_timeout` - See Argument Reference above.
- `tcp_draining` - See Argument Reference above.
- `tcp_draining_timeout` - See Argument Reference above.
- `certificate_id` - See Argument Reference above.
- `udp_timeout` - See Argument Reference above.
- `ssl_protocols` - See Argument Reference above.
- `ssl_ciphers` - See Argument Reference above.
- `id` - Specifies the listener ID.
- `admin_state_up` - Specifies the status of the load balancer. Value range: `false`: The load balancer is disabled. `true`: The load balancer runs properly.

flexibleengine_elb_loadbalancer

Manages an elastic loadbalancer resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_elb_loadbalancer" "elb" {
  name = "elb"
  type = "External"
  description = "test elb"
  vpc_id = "e346dc4a-d9a6-46f4-90df-10153626076e"
  admin_state_up = 1
  bandwidth = 5
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the loadbalancer. If omitted, the `region` argument of the provider is used. Changing this creates a new loadbalancer.
- `name` - (Required) Specifies the load balancer name. The name is a string of 1 to 64 characters that consist of letters, digits, underscores (`_`), and hyphens (`-`).
- `description` - (Optional) Provides supplementary information about the listener. The value is a string of 0 to 128 characters and cannot be `<>`.
- `vpc_id` - (Required) Specifies the VPC ID.
- `bandwidth` - (Optional) Specifies the bandwidth (Mbit/s). This parameter is mandatory when `type` is set to `External`, and it is invalid when `type` is set to `Internal`. The value ranges from 1 to 300.
- `type` - (Required) Specifies the load balancer type. The value can be `Internal` or `External`.
- `admin_state_up` - (Required) Specifies the status of the load balancer. Value range: 0 or false: indicates that the load balancer is stopped. Only tenants are allowed to enter these two values. 1 or true: indicates that the load balancer is running properly. 2 or false: indicates that the load balancer is frozen. Only tenants are allowed to enter these two values.
- `vip_subnet_id` - (Optional) Specifies the ID of the private network to be added. This parameter is mandatory when `type` is set to `Internal`, and it is invalid when `type` is set to `External`.
- `az` - (Optional) Specifies the ID of the availability zone (AZ). This parameter is mandatory when `type` is set to `Internal`, and it is invalid when `type` is set to `External`.
- `security_group_id` - (Optional) Specifies the security group ID. The value is a string of 1 to 200 characters that consists of uppercase and lowercase letters, digits, and hyphens (`-`). This parameter is mandatory only when `type` is set to `Internal`.

- `vip_address` - (Optional) Specifies the IP address provided by ELB. When `type` is set to External, the value of this parameter is the elastic IP address. When `type` is set to Internal, the value of this parameter is the private network IP address. You can select an existing elastic IP address and create a public network load balancer. When this parameter is configured, parameter `bandwidth` is invalid.
- `tenant_id` - (Optional) Specifies the tenant ID. This parameter is mandatory only when `type` is set to Internal.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `vpc_id` - See Argument Reference above.
- `bandwidth` - See Argument Reference above.
- `type` - See Argument Reference above.
- `admin_state_up` - See Argument Reference above.
- `vip_subnet_id` - See Argument Reference above.
- `az` - See Argument Reference above.
- `security_group_id` - See Argument Reference above.
- `vip_address` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `id` - Specifies the load balancer ID.

flexibleengine_fw_firewall_group_v2

Manages a v2 firewall group resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_fw_rule_v2" "rule_1" {
  name           = "my-rule-1"
  description    = "drop TELNET traffic"
  action         = "deny"
  protocol       = "tcp"
  destination_port = "23"
  enabled        = "true"
}

resource "flexibleengine_fw_rule_v2" "rule_2" {
  name           = "my-rule-2"
  description    = "drop NTP traffic"
  action         = "deny"
  protocol       = "udp"
  destination_port = "123"
  enabled        = "false"
}

resource "flexibleengine_fw_policy_v2" "policy_1" {
  name = "my-policy"

  rules = [
    "${flexibleengine_fw_rule_v2.rule_1.id}",
    "${flexibleengine_fw_rule_v2.rule_2.id}",
  ]
}

resource "flexibleengine_fw_firewall_group_v2" "firewall_group_1" {
  name           = "my-firewall-group"
  ingress_policy_id = "${flexibleengine_fw_policy_v2.policy_1.id}"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the v2 networking client. A networking client is needed to create a firewall group. If omitted, the `region` argument of the provider is used. Changing this creates a new firewall group.
- `ingress_policy_id` - The ingress policy resource id for the firewall group. Changing this updates the `ingress_policy_id` of an existing firewall group.
- `egress_policy_id` - The egress policy resource id for the firewall group. Changing this updates the `egress_policy_id` of an existing firewall group.
- `name` - (Optional) A name for the firewall group. Changing this updates the `name` of an existing firewall group.

- `description` - (Required) A description for the firewall group. Changing this updates the `description` of an existing firewall group.
- `admin_state_up` - (Optional) Administrative up/down status for the firewall group (must be "true" or "false" if provided - defaults to "true"). Changing this updates the `admin_state_up` of an existing firewall group.
- `tenant_id` - (Optional) The owner of the floating IP. Required if admin wants to create a firewall group for another tenant. Changing this creates a new firewall group.
- `ports` - (Optional) Port(s) to associate this firewall group instance with. Must be a list of strings. Changing this updates the associated routers of an existing firewall group.
- `value_specs` - (Optional) Map of additional options.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `policy_id` - See Argument Reference above.
- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `admin_state_up` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `ports` - See Argument Reference above.

Import

Firewall Groups can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_fw_firewall_group_v2.firewall_group_1 c9e39fb2-ce20-46c8-a964-25f3898c7a97
```

flexibleengine_fw_policy_v2

Manages a v2 firewall policy resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_fw_rule_v2" "rule_1" {
  name          = "my-rule-1"
  description   = "drop TELNET traffic"
  action        = "deny"
  protocol      = "tcp"
  destination_port = "23"
  enabled       = "true"
}

resource "flexibleengine_fw_rule_v2" "rule_2" {
  name          = "my-rule-2"
  description   = "drop NTP traffic"
  action        = "deny"
  protocol      = "udp"
  destination_port = "123"
  enabled       = "false"
}

resource "flexibleengine_fw_policy_v2" "policy_1" {
  name = "my-policy"

  rules = [
    "${flexibleengine_fw_rule_v2.rule_1.id}",
    "${flexibleengine_fw_rule_v2.rule_2.id}",
  ]
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the v2 networking client. A networking client is needed to create a firewall policy. If omitted, the `region` argument of the provider is used. Changing this creates a new firewall policy.
- `name` - (Optional) A name for the firewall policy. Changing this updates the `name` of an existing firewall policy.
- `description` - (Optional) A description for the firewall policy. Changing this updates the `description` of an existing firewall policy.
- `rules` - (Optional) An array of one or more firewall rules that comprise the policy. Changing this results in adding/removing rules from the existing firewall policy.
- `audited` - (Optional) Audit status of the firewall policy (must be "true" or "false" if provided - defaults to "false"). This status is set to "false" whenever the firewall policy or any of its rules are changed. Changing this updates the `audited` status of an existing firewall policy.

- `shared` - (Optional) Sharing status of the firewall policy (must be "true" or "false" if provided). If this is "true" the policy is visible to, and can be used in, firewalls in other tenants. Changing this updates the `shared` status of an existing firewall policy. Only administrative users can specify if the policy should be shared.
- `value_specs` - (Optional) Map of additional options.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `audited` - See Argument Reference above.
- `shared` - See Argument Reference above.

Import

Firewall Policies can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_fw_policy_v2.policy_1 07f422e6-c596-474b-8b94-fe2c12506ce0
```

flexibleengine_fw_rule_v2

Manages a v2 firewall rule resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_fw_rule_v2" "rule_1" {
  name          = "my_rule"
  description   = "drop TELNET traffic"
  action        = "deny"
  protocol      = "tcp"
  destination_port = "23"
  enabled       = "true"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the v2 networking client. A Compute client is needed to create a firewall rule. If omitted, the `region` argument of the provider is used. Changing this creates a new firewall rule.
- `name` - (Optional) A unique name for the firewall rule. Changing this updates the `name` of an existing firewall rule.
- `description` - (Optional) A description for the firewall rule. Changing this updates the `description` of an existing firewall rule.
- `protocol` - (Required) The protocol type on which the firewall rule operates. Valid values are: `tcp`, `udp`, `icmp`, and `any`. Changing this updates the `protocol` of an existing firewall rule.
- `action` - (Required) Action to be taken (must be "allow" or "deny") when the firewall rule matches. Changing this updates the `action` of an existing firewall rule.
- `ip_version` - (Optional) IP version, either 4 (default) or 6. Changing this updates the `ip_version` of an existing firewall rule.
- `source_ip_address` - (Optional) The source IP address on which the firewall rule operates. Changing this updates the `source_ip_address` of an existing firewall rule.
- `destination_ip_address` - (Optional) The destination IP address on which the firewall rule operates. Changing this updates the `destination_ip_address` of an existing firewall rule.
- `source_port` - (Optional) The source port on which the firewall rule operates. Changing this updates the `source_port` of an existing firewall rule.
- `destination_port` - (Optional) The destination port on which the firewall rule operates. Changing this updates the `destination_port` of an existing firewall rule.
- `enabled` - (Optional) Enabled status for the firewall rule (must be "true" or "false" if provided - defaults to "true"). Changing this updates the `enabled` status of an existing firewall rule.

- `tenant_id` - (Optional) The owner of the firewall rule. Required if admin wants to create a firewall rule for another tenant. Changing this creates a new firewall rule.
- `value_specs` - (Optional) Map of additional options.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `protocol` - See Argument Reference above.
- `action` - See Argument Reference above.
- `ip_version` - See Argument Reference above.
- `source_ip_address` - See Argument Reference above.
- `destination_ip_address` - See Argument Reference above.
- `source_port` - See Argument Reference above.
- `destination_port` - See Argument Reference above.
- `enabled` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.

Import

Firewall Rules can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_fw_rule_v2.rule_1 8dbc0c28-e49c-463f-b712-5c5d1bbac327
```

flexibleengine_images_image_v2

Manages a V2 Image resource within FlexibleEngine Glance.

Example Usage

```
resource "flexibleengine_images_image_v2" "rancheros" {
  name      = "RancherOS"
  image_source_url = "https://releases.rancher.com/os/latest/rancheros-openstack.img"
  container_format = "bare"
  disk_format = "qcow2"
}
```

Argument Reference

The following arguments are supported:

- `container_format` - (Required) The container format. Must be one of "ami", "ari", "aki", "bare", "ovf".
- `disk_format` - (Required) The disk format. Must be one of "ami", "ari", "aki", "vhd", "vmdk", "raw", "qcow2", "vdi", "iso".
- `local_file_path` - (Optional) This is the filepath of the raw image file that will be uploaded to Glance. Conflicts with `image_source_url`.
- `image_cache_path` - (Optional) This is the directory where the images will be downloaded. Images will be stored with a filename corresponding to the url's md5 hash. Defaults to "\$HOME/.terraform/image_cache"
- `image_source_url` - (Optional) This is the url of the raw image that will be downloaded in the `image_cache_path` before being uploaded to Glance. Glance is able to download image from internet but the `gophercloud` library does not yet provide a way to do so. Conflicts with `local_file_path`.
- `min_disk_gb` - (Optional) Amount of disk space (in GB) required to boot image. Defaults to 0.
- `min_ram_mb` - (Optional) Amount of ram (in MB) required to boot image. Defaults to 0.
- `name` - (Required) The name of the image.
- `protected` - (Optional) If true, image will not be deletable. Defaults to false.
- `region` - (Optional) The region in which to obtain the V2 Glance client. A Glance client is needed to create an Image that can be used with a compute instance. If omitted, the `region` argument of the provider is used. Changing this creates a new Image.
- `tags` - (Optional) The tags of the image. It must be a list of strings. At this time, it is not possible to delete all tags of an image.
- `visibility` - (Optional) The visibility of the image. Must be one of "public", "private", "community", or "shared". The ability to set the visibility depends upon the configuration of the FlexibleEngine cloud.

Note: The `properties` attribute handling in the `gophercloud` library is currently buggy and needs to be fixed before being implemented in this resource.

Attributes Reference

The following attributes are exported:

- `checksum` - The checksum of the data associated with the image.
- `container_format` - See Argument Reference above.
- `created_at` - The date the image was created.
- `disk_format` - See Argument Reference above.
- `file` - the trailing path after the glance endpoint that represent the location of the image or the path to retrieve it.
- `id` - A unique ID assigned by Glance.
- `metadata` - The metadata associated with the image. Image metadata allow for meaningfully define the image properties and tags. See <http://docs.openstack.org/developer/glance/metadefs-concepts.html> (<http://docs.openstack.org/developer/glance/metadefs-concepts.html>).
- `min_disk_gb` - See Argument Reference above.
- `min_ram_mb` - See Argument Reference above.
- `name` - See Argument Reference above.
- `owner` - The id of the flexibleengine user who owns the image.
- `protected` - See Argument Reference above.
- `region` - See Argument Reference above.
- `schema` - The path to the JSON-schema that represent the image or image
- `size_bytes` - The size in bytes of the data associated with the image.
- `status` - The status of the image. It can be "queued", "active" or "saving".
- `tags` - See Argument Reference above.
- `update_at` - The date the image was last updated.
- `visibility` - See Argument Reference above.

Import

Images can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_images_image_v2.rancheros 89c60255-9bd6-460c-822a-e2b959ede9d2
```

flexibleengine_kms_key_v1

Manages a V1 key resource within KMS.

Example Usage

```
resource "flexibleengine_kms_key_v1" "key_1" {
  key_alias      = "key_1"
  pending_days  = "7"
  key_description = "first test key"
  realm         = "cn-north-1"
  is_enabled    = true
}
```

Argument Reference

The following arguments are supported:

- `key_alias` - (Required) The alias in which to create the key. It is required when we create a new key. Changing this updates the alias of key.
- `key_description` - (Optional) The description of the key as viewed in FlexibleEngine console. Changing this updates the description of key.
- `realm` - (Optional) Region where a key resides. Changing this creates a new key.
- `pending_days` - (Optional) Duration in days after which the key is deleted after destruction of the resource, must be between 7 and 1096 days. Defaults to 7. It only be used when delete a key.
- `is_enabled` - (Optional) Specifies whether the key is enabled. Defaults to true. Changing this updates the state of existing key.

Attributes Reference

The following attributes are exported:

- `key_alias` - See Argument Reference above.
- `key_description` - See Argument Reference above.
- `realm` - See Argument Reference above.
- `key_id` - The globally unique identifier for the key.
- `default_key_flag` - Identification of a Master Key. The value 1 indicates a Default Master Key, and the value 0 indicates a key.
- `origin` - Origin of a key. The default value is kms.

- `scheduled_deletion_date` - Scheduled deletion time (time stamp) of a key.
- `domain_id` - ID of a user domain for the key.
- `expiration_time` - Expiration time.
- `creation_date` - Creation time (time stamp) of a key.
- `is_enabled` - See Argument Reference above.

Import

KMS Keys can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_kms_key_v1.key_1 7056d636-ac60-4663-8a6c-82d3c32c1c64
```

flexibleengine_lb_certificate_v2

Manages a V2 certificate resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_lb_certificate_v2" "certificate_1" {
  name = "certificate_1"
  description = "terraform test certificate"
  domain = "www.elb.com"
  private_key = <<EOT
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAACAQEAwZ5UJULAJwR7p6FVwGRQRjFN2s8tZ/6LC3X82fajpVsYqF1x
qEuUDndDXVD09E4u83MS6H06a3bIVQDp6/klnYldiE6Vp8HH5BSKaCWKvg8lGwg1
UM9wZFnLryi14KgmPIFmcu9nA8yV/6MZAe6RSDmb3iyNBmiZ8aZhGw2pI1YwR+15
MVqFFGB+7ExkziR0i7L8FCyCezK2/o00vQsH1dzQ8z1JXWdg8/9Zx7Ktvgwu5PQ
M3cJtSHX6iBP0kMU8Z8TugLLtQqXKZ0EgwajwvQ5mf2DPkVgM08XAgALJcLigwD5
13koAdtJd5v+9irw+5LAu03JclqwTvwv7u/YwwIDAQABAoIBACU9S5fjD9/jTMA
DRs08A+gGgZUXLn0xk+NAPX3LyB1tfdkCaFB8BccLz06h3KZuwQOBpv6jkdvEDbx
Nwyw3eA/9GJsIvKiHc0rejdvypymaw9I8MA7NbXHaJrY7KpQDQyk6sx+aUTCy5jg
iMXLWdwXYHhJ/1HV0o603oZyiS6HZeYU089NDUCx+1Sji3e5Ke0gPVXEqCq1011/
rh24bMxnwZ04PKBwdcMBN5Zf/4ij9vrZE+ffZw7vGB048A5lvZxWU2U5t/OZQRtN
1uLOHmMFA0FIF2aWbTVfwdUWAFsvA0kHj9VV8BX0UwK0UuEktdkFAlvrxmsFr0/H
yDeYYPkCgYEA/S55CBbR0sMXpSZ56uRn8JHApZJhgkgvYr+FqDLJq/e92nAzf01P
RoEBUajwrnf1ycevN/SDfbtWzq2XJGqHwDjmtP016b7KBsC6BdRcH6dn0Yh31jgA
vABMIP3wzI4zSVTyxRE8LDuboytF1mSCeV5tHYPQTZNwrpLDnLQhywcGyEAw8Yc
Uk/eiFr3hfH/ZohMfV5p82Qp7DNIGRzw8YtVG/3+vNXrAXW1VhugNhQY6L+zLJC
aKn84ooup0m3YCg0hVINqJuvzfsuzQgtjTXyaE0cEwsjUus0miuj09vVx/3U7siK
Hdj2ICPCvQ6Q8tdi8jV320gMs05AtaBkZdsiWUCgYEAtLw4Kk4f+xTKDFsrLUNf
75wcqhWVBiwBp7yQ7UX4EYSJPKZcHMRTk0EEcAbpyaJZE3I44vjp5ReXIHNLmFps
uvI34J4Rfot0LN3n7cFrAi2+wpNo+MOBwrNzprMijGP2uKkrq4JiMjFbKV/6utGF
Up7VxfwS904JYpqGaZctiIECGYA1A6nZtF0riY6ry/uAdXpZHL8ONNqRZtWoT0kD
79otSVu5ISiRbaGcXsDExC52oKrSDAgFtbqQuiE0Fg09UcXfoR6HwRkba2CiDwve
yHQLQI5Qrdxz8Mk0gIrnRSM4FamcW9vi9z4kCbQyoC5C+4gqeULJRpDIkQBWP2Y4
2ct/bQKBgHv8qCsQTzph0xc31BJPa2xVhuv18cEU3XLURvFUZ/1f43JhLp7gynS2
ep++LKUiu9D0VGXY8bqvFjJbECoCeu85v18NpCXwe/LoVoIn+7KaVIZMwqoGMfgNL
nEqm7HWkNxxHhF8A6En/IjleuddS1sf9e/x+TJN1Xhnt9W6pe7Fk1
-----END RSA PRIVATE KEY-----
EOT

certificate = <<EOT
-----BEGIN CERTIFICATE-----
MIIDpTCCAO2gAwIBAgIJAKdmm0BYnFvoMA0GCSqGSIb3DQEBCwUAMGkxCzAJBgNV
BAYTAnh4MQswCQYDVQQLDAJ4eDELMAKGA1UEBwwCeHgxZCZAJBgNVBAoManh4MQsw
CQYDVQQLDAJ4eDELMAKGA1UEAwwCeHgxGTAXBgkqhkiG9w0BCQEWCnh4QDE2My5j
b20wHhcNMjcxMjA0MDM0MjQ5MjYwMjYwMjYwMjYwMjYwMjYwMjYwMjYwMjYwMjYw
eDELMAKGA1UECwAwCeHgxZCZAJBgNVBAMManh4MQswCQYDVQQLDAJ4eDELMAKGA1UE
CwwCeHgxZCZAJBgNVBAMManh4MRkwFwYJKoZIhvcNAQkBFgpc4eEAXNjMuY29tMIIIB
IjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAwZ5UJULAJwR7p6FVwGRQRjFN
2s8tZ/6LC3X82fajpVsYqF1xqEuUDndDXVD09E4u83MS6H06a3bIVQDp6/klnYld
iE6Vp8HH5BSKaCWKvg8lGwg1UM9wZFnLryi14KgmPIFmcu9nA8yV/6MZAe6RSDmb
3iyNBmiZ8aZhGw2pI1YwR+15MVqFFGB+7ExkziR0i7L8FCyCezK2/o00vQsH1dz
Q8z1JXWdg8/9Zx7Ktvgwu5PQM3cJtSHX6iBP0kMU8Z8TugLLtQqXKZ0EgwajwvQ5
mf2DPkVgM08XAgALJcLigwD513koAdtJd5v+9irw+5LAu03JclqwTvwv7u/YwwID
AQABo1AwTjAdBgNVHQ4EFgQUo5A2tIu+bcUfvGTD7wmEkhXKfjcwHwYDVR0jBBGw
```

```
FoAUo5A2tIu+bcUfvGTD7wmEkhXKFjcwDAYDVR0TBAAUwAwEB/zANBgkqhkiG9w0B
AQsFAA0CAQEAWJ2rS6MvLqk3GfEpboezx2J3X7L1z8Sxoqg6ntwB+rezvK3mc9H0
83qcVeUcoH+0A0LSHyFN4FvRQL6X1hEheHarYwJK4agb231vb5erasuG0463eYEG
r4SfTu0m7SyiV2xxbaBKrXJtpBp4WLL/s+LF+nklKja0xkmxUX0sM4CTA7uFJypY
c8Tdr8LDDNqoUtMD8BrUCJi+7lmMXRcC3Qi3oZJW76ja+kZA5mKVFPd1ATih8TbA
i34R7EQDtFeiSvBdeKRSPp8c0KT8H1B4LXNkkCQs2WX5p4lm99+ZtLD4glw8x6Ic
i1YhgnQbn5E0hz550Lu5jv0kKQjPCW+8Kg==
```

-----END CERTIFICATE-----

EOT

```
timeouts {
  create = "5m"
  update = "5m"
  delete = "5m"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create an LB certificate. If omitted, the `region` argument of the provider is used. Changing this creates a new LB certificate.
- `name` - (Optional) Human-readable name for the Certificate. Does not have to be unique.
- `description` - (Optional) Human-readable description for the Certificate.
- `domain` - (Optional) The domain of the Certificate.
- `private_key` - (Required) The private encrypted key of the Certificate, PEM format.
- `certificate` - (Required) The public encrypted key of the Certificate, PEM format.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `domain` - See Argument Reference above.
- `private_key` - See Argument Reference above.
- `certificate` - See Argument Reference above.
- `update_time` - Indicates the update time.
- `create_time` - Indicates the creation time.

flexibleengine_lb_l7policy_v2

Manages a Load Balancer L7 Policy resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_lb_loadbalancer_v2" "loadbalancer_1" {
  name = "loadbalancer_1"
  vip_subnet_id = "SUBNET_ID"
}

resource "flexibleengine_lb_listener_v2" "listener_1" {
  name = "listener_1"
  protocol = "HTTP"
  protocol_port = 8080
  loadbalancer_id = "${flexibleengine_lb_loadbalancer_v2.loadbalancer_1.id}"
}

resource "flexibleengine_lb_pool_v2" "pool_1" {
  name          = "pool_1"
  protocol      = "HTTP"
  lb_method     = "ROUND_ROBIN"
  loadbalancer_id = "${flexibleengine_lb_loadbalancer_v2.loadbalancer_1.id}"
}

resource "flexibleengine_lb_l7policy_v2" "l7policy_1" {
  name          = "test"
  action        = "REDIRECT_TO_POOL"
  description   = "test l7 policy"
  position      = 1
  listener_id   = "${flexibleengine_lb_listener_v2.listener_1.id}"
  redirect_pool_id = "${flexibleengine_lb_pool_v2.pool_1.id}"
}
```

Argument Reference

The following arguments are supported:

- **region** - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create an L7 Policy. If omitted, the `region` argument of the provider is used. Changing this creates a new L7 Policy.
- **tenant_id** - (Optional) Required for admins. The UUID of the tenant who owns the L7 Policy. Only administrative users can specify a tenant UUID other than their own. Changing this creates a new L7 Policy.
- **name** - (Optional) Human-readable name for the L7 Policy. Does not have to be unique.
- **description** - (Optional) Human-readable description for the L7 Policy.
- **action** - (Required) The L7 Policy action - can either be `REDIRECT_TO_POOL`, or `REDIRECT_TO_LISTENER`. Changing this creates a new L7 Policy.

- `listener_id` - (Required) The Listener on which the L7 Policy will be associated with. Changing this creates a new L7 Policy.
- `position` - (Optional) The position of this policy on the listener. Positions start at 1. Changing this creates a new L7 Policy.
- `redirect_pool_id` - (Optional) Requests matching this policy will be redirected to the pool with this ID. Only valid if action is `REDIRECT_TO_POOL`.
- `redirect_listener_id` - (Optional) Requests matching this policy will be redirected to the listener with this ID. Only valid if action is `REDIRECT_TO_LISTENER`.
- `admin_state_up` - (Optional) The administrative state of the L7 Policy. This value can only be true (UP).

Attributes Reference

The following attributes are exported:

- `id` - The unique ID for the L7 Policy.
- `region` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `action` - See Argument Reference above.
- `listener_id` - See Argument Reference above.
- `position` - See Argument Reference above.
- `redirect_pool_id` - See Argument Reference above.
- `redirect_listener_id` - See Argument Reference above.
- `admin_state_up` - See Argument Reference above.

Import

Load Balancer L7 Policy can be imported using the L7 Policy ID, e.g.:

```
$ terraform import flexibleengine_lb_l7policy_v2.l7policy_1 8a7a79c2-cf17-4e65-b2ae-ddc8bfcf6c74
```

flexibleengine_lb_l7rule_v2

Manages a V2 L7 Rule resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_lb_loadbalancer_v2" "loadbalancer_1" {
  name = "loadbalancer_1"
  vip_subnet_id = "SUBNET_ID"
}

resource "flexibleengine_lb_listener_v2" "listener_1" {
  name = "listener_1"
  protocol = "HTTP"
  protocol_port = 8080
  loadbalancer_id = "${flexibleengine_lb_loadbalancer_v2.loadbalancer_1.id}"
}

resource "flexibleengine_lb_pool_v2" "pool_1" {
  name          = "pool_1"
  protocol      = "HTTP"
  lb_method     = "ROUND_ROBIN"
  loadbalancer_id = "${flexibleengine_lb_loadbalancer_v2.loadbalancer_1.id}"
}

resource "flexibleengine_lb_l7policy_v2" "l7policy_1" {
  name          = "test"
  action        = "REDIRECT_TO_URL"
  description   = "test description"
  position     = 1
  listener_id   = "${flexibleengine_lb_listener_v2.listener_1.id}"
  redirect_url  = "http://www.example.com"
}

resource "flexibleengine_lb_l7rule_v2" "l7rule_1" {
  l7policy_id = "${flexibleengine_lb_l7policy_v2.l7policy_1.id}"
  type        = "PATH"
  compare_type = "EQUAL_TO"
  value       = "/api"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create an L7 Rule. If omitted, the `region` argument of the provider is used. Changing this creates a new L7 Rule.
- `tenant_id` - (Optional) Required for admins. The UUID of the tenant who owns the L7 Rule. Only administrative users can specify a tenant UUID other than their own. Changing this creates a new L7 Rule.

- `description` - (Optional) Human-readable description for the L7 Rule.
- `type` - (Required) The L7 Rule type - can either be `HOST_NAME` or `PATH`. Changing this creates a new L7 Rule.
- `compare_type` - (Required) The comparison type for the L7 rule - can either be `STARTS_WITH`, `EQUAL_TO` or `REGEX`
- `l7policy_id` - (Required) The ID of the L7 Policy to query. Changing this creates a new L7 Rule.
- `value` - (Required) The value to use for the comparison. For example, the file type to compare.
- `key` - (Optional) The key to use for the comparison. For example, the name of the cookie to evaluate. Valid when `type` is set to `COOKIE` or `HEADER`. Changing this creates a new L7 Rule.
- `admin_state_up` - (Optional) The administrative state of the L7 Rule. The value can only be `true` (UP).

Attributes Reference

The following attributes are exported:

- `id` - The unique ID for the L7 Rule.
- `region` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `type` - See Argument Reference above.
- `compare_type` - See Argument Reference above.
- `l7policy_id` - See Argument Reference above.
- `value` - See Argument Reference above.
- `key` - See Argument Reference above.
- `invert` - See Argument Reference above.
- `admin_state_up` - See Argument Reference above.
- `listener_id` - The ID of the Listener owning this resource.

Import

Load Balancer L7 Rule can be imported using the L7 Policy ID and L7 Rule ID separated by a slash, e.g.:

```
$ terraform import flexibleengine_lb_l7rule_v2.l7rule_1 e0bd694a-abbe-450e-b329-0931fd1cc5eb/4086b0c9-b18c-4d1c-b6b8-4c56c3ad2a9e
```

flexibleengine_lb_listener_v2

Manages a V2 listener resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_lb_listener_v2" "listener_1" {
  protocol      = "HTTP"
  protocol_port = 8080
  loadbalancer_id = "d9415786-5f1a-428b-b35f-2f1523e146d2"
}
```

Example Usage of TERMINATED_HTTPS protocol

```
resource "flexibleengine_lb_loadbalancer_v2" "loadbalancer_1" {
  name = "loadbalancer_cert"
  vip_subnet_id = "2c0a74a9-4395-4e62-a17b-e3e86fbf66b7"
}
```

```
resource "flexibleengine_lb_certificate_v2" "certificate_1" {
  name = "cert"
  domain = "www.elb.com"
  private_key = <<EOT
-----BEGIN RSA PRIVATE KEY-----
MIIIEowIBAAKCAQEAWZ5UJULAJWr7p6FVwGRQRjFN2s8tZ/6LC3X82faJpVsYqF1x
qEuUDndDXVD09E4u83MS6HO6a3bIVQDp6/klnYldiE6Vp8HH5BSKaCWKvg8LGWg1
UM9wZFnLryi14KgmPIFmcu9nA8yV/6MZAe6RSDmb3iyNBmiZ8aZhGw2pI1YwR+15
MVqFFGB+7ExkziR0i7L8CFCyCezK2/o00vQsH1dzQ8z1JXWdg8/9Zx7Ktvgwu5PQ
M3cJtSHX6iBP0kMU8Z8TugLLTqQXKZ0EgwaJwvQ5mf2DPkVgM08XAgALJcLigWd5
13koAdtJd5v+9irw+5LAu03JclqwTvwY7u/YwwIDAQABAoIBACU9S5fjd9/jTmXA
DRs08A+gGgZUXLn0xk+NAPX3LyB1tfdkCaFB8BccLz06h3KZuwQOBpV6jkdvEDbx
Nwyw3eA/9GJsIvKiHc0rejdvypymaw9I8MA7NbXHaJrY7KpqDQyk6sx+aUTcy5jg
iMxLWdwXYHhJ/1HV0o603oZyiS6HZeYU089NDUCx+1Sji3e5Ke0gPVXEqCq1011/
rh24bMxnwZo4PKBwdcMBN5Zf/4ij9vrZE+fFzW7vGB048A5lvZxWU2U5t/OZQRtN
1uLOHmMfa0FIF2aWbTVfwdUwAFsvA0kHj9VV8BX0UwK0UuEktdkfAlvrXmsFr0/H
yDeYYPkCgYEA/S55CBbR0sMXPsz56uRn8JHApZJhgkgvYr+FqDlJq/e92nAzf01P
RoEBUajwrnf1ycevN/SDfbtWzq2XJGqhWdJmtP016b7KbsC6BdRcH6dn0Yh31jgA
vABMIP3wzI4zSVTyxRE8LDuboytF1mSceV5tHYPQTZNwrplDnLQhywcGyEAw8Yc
Uk/eiFr3hfH/ZohMfV5p82Qp7DNIGRzw8YtVG/3+vNXrAXW1VhugNhQY6L+zLtJC
aKn84ooup0m3YcG0hVINqJuvzfsuzQgtjTXyaE0cEwsjUus0miuj09vVx/3U7siK
Hdj2ICPcVQ6Q8tdi8jV320gMs05AtaBkZdsiWUCgYEAtLw4Kk4f+xTKDFsrLUNf
75wcqhWVBiwBp7yQ7UX4EYsJPKZcHMRTk0EEcAbpyaJZE3I44vj5p5ReXIHNLmfPs
uvI34J4Rfot0LN3n7cFrAi2+wpNo+MOBwrNzprMijGP2uKKrQ4JiMjFbKV/6utGF
Up7VxfwS904JYpqGaZctiIECgYA1A6nZtF0riY6ry/uAdXpZHL8ONNqRZtWoT0kD
79otSVu5ISiRbaGcXsDEXc52oKrSDAgFtbqUiE0Fg09UcXfoR6HwRkba2CiDwve
yHQLQI5Qrdxz8Mk0gIrrNrSM4FamcW9vi9z4kCbQyoc5C+4gqeULJRpDIkQBWP2Y4
2ct/bQKBgHv8qCsQTzph0xc31BJPa2xVhuv18cEU3XLUrVfUZ/1f43JhLp7gynS2
ep++LKUI9D0VGXY8bqvFjJbECoCeu85v18NpCXwe/LoVoIn+7KaVIZMwqoGMfgNL
nEqm7HWkNxxHh8A6En/IjleuddS1sf9e/x+TJN1Xhnt9W6pe7Fk1
-----END RSA PRIVATE KEY-----
EOT
```

```

certificate = <<EOT
-----BEGIN CERTIFICATE-----
MIIDpTCCAo2gAwIBAgIJAKdmm0BYnFvoMA0GCSqGSIb3DQEBCwUAMGkxCzAJBgNV
BAYTAnh4MQswCQYDVQQLIDAJ4eDELMAkGA1UEBwwCeHgxZCZAJBgNVBAAoMAnh4MQsw
CQYDVQQLDAJ4eDELMAkGA1UEAwwCeHgxGTAXBgkqhkiG9w0BCQEWEnh4QDE2My5j
b20wHhcNMTCxMjA0MDM0MjQ5WWhcNMjAxMjAzMDM0MjQ5WjBpMQswCQYDVQQLGEwJ4
eDELMAkGA1UECAwCeHgxZCZAJBgNVBACMAAnh4MQswCQYDVQKDAJ4eDELMAkGA1UE
CwwCeHgxZCZAJBgNVBAMMAAnh4MRkwFwYJKoZIhvcNAQkBFgp4eEAXNjMuY29tMIIB
IjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAwZ5UJULAJwR7p6FVwGRQRjFN
2s8tZ/6LC3X82fajpVsYqF1xqEuUDndDXVD09E4u83MS6H06a3bIVQDp6/klnYld
iE6Vp8HH5BSKaCWKVg8lGWg1UM9wZFnlryi14KgmpIFmCu9nA8yV/6MZAe6RSDmb
3iyNBmiZ8aZhGw2pI1YwR+15MVqFFGB+7ExkziR0i7L8CFyCezK2/o00vQsH1dz
Q8z1JXWdg8/9Zx7Ktvgwu5PQM3cJtSHX6iBP0kMU8Z8TugLlTqQXKZOEgwaJwvQ5
mf2DPkVgM08XAgALJcLigwD513koAdtJd5v+9irw+5LAu03JclqwTvvy7u/YwwID
AQABo1AwTjAdBgNVHQ4EFgQUo5A2tIu+bcUfvGTD7wmEkhXKFjcwHwYDVR0jBBgw
FoAUo5A2tIu+bcUfvGTD7wmEkhXKFjcwDAYDVR0TBAAUwAwEB/zANBgkqhkiG9w0B
AQsFAAOCAQEAJ2rS6Mvlqk3GfEpboezx2J3X7l1z8Sxoqg6ntwB+rezvK3mc9H0
83qcVeUcoH+0A0lSHyFN4FvRQL6X1hEheHarYwJK4agb231vb5erasuG0463eYEG
r4SfTu0m7SyiV2xxbaBKrXJtpBp4WLL/s+LF+nklKja0xkxUX0sM4CTA7uFJypY
c8Tdr8lDDNqoUtMD8BrUCJi+7lmMXRcC3Qi3oZJW76ja+kZA5mKVFPd1ATih8TbA
i34R7EQDtFeiSvBdeKRSPp8c0KT8H1B4LXNkkCQs2WX5p4lm99+ZtLD4glw8x6Ic
i1YhgnQbn5E0hz550Lu5jv0kKqjPCW+9Aa==
-----END CERTIFICATE-----
EOT
}

resource "flexibleengine_lb_listener_v2" "listener_1" {
  name = "listener_cert"
  protocol = "TERMINATED_HTTPS"
  protocol_port = 8080
  loadbalancer_id = "${flexibleengine_lb_loadbalancer_v2.loadbalancer_1.id}"
  default_tls_container_ref = "${flexibleengine_lb_certificate_v2.certificate_1.id}"
}

```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create an . If omitted, the `region` argument of the provider is used. Changing this creates a new Listener.
- `protocol` - (Required) The protocol - can either be TCP, UDP, HTTP or TERMINATED_HTTPS. Changing this creates a new Listener.
- `protocol_port` - (Required) The port on which to listen for client traffic. Changing this creates a new Listener.
- `tenant_id` - (Optional) Required for admins. The UUID of the tenant who owns the Listener. Only administrative users can specify a tenant UUID other than their own. Changing this creates a new Listener.
- `loadbalancer_id` - (Required) The load balancer on which to provision this Listener. Changing this creates a new Listener.
- `name` - (Optional) Human-readable name for the Listener. Does not have to be unique.

- `default_pool_id` - (Optional) The ID of the default pool with which the Listener is associated. Changing this creates a new Listener.
- `description` - (Optional) Human-readable description for the Listener.
- `default_tls_container_ref` - (Optional) A reference to a Barbican Secrets container which stores TLS information. This is required if the protocol is `TERMINATED_HTTPS`. See here (<https://wiki.openstack.org/wiki/Network/LBaaS/docs/how-to-create-tls-loadbalancer>) for more information.
- `sni_container_refs` - (Optional) A list of references to Barbican Secrets containers which store SNI information. See here (<https://wiki.openstack.org/wiki/Network/LBaaS/docs/how-to-create-tls-loadbalancer>) for more information.
- `admin_state_up` - (Optional) The administrative state of the Listener. A valid value is `true` (UP) or `false` (DOWN).

Attributes Reference

The following attributes are exported:

- `id` - The unique ID for the Listener.
- `protocol` - See Argument Reference above.
- `protocol_port` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `name` - See Argument Reference above.
- `default_port_id` - See Argument Reference above.
- `description` - See Argument Reference above.
- `connection_limit` - See Argument Reference above.
- `default_tls_container_ref` - See Argument Reference above.
- `sni_container_refs` - See Argument Reference above.
- `admin_state_up` - See Argument Reference above.

flexibleengine_lb_loadbalancer_v2

Manages a V2 loadbalancer resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_lb_loadbalancer_v2" "lb_1" {  
  vip_subnet_id = "d9415786-5f1a-428b-b35f-2f1523e146d2"  
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create an LB member. If omitted, the `region` argument of the provider is used. Changing this creates a new LB member.
- `vip_subnet_id` - (Required) The network on which to allocate the Loadbalancer's address. A tenant can only create Loadbalancers on networks authorized by policy (e.g. networks that belong to them or networks that are shared). Changing this creates a new loadbalancer.
- `name` - (Optional) Human-readable name for the Loadbalancer. Does not have to be unique.
- `description` - (Optional) Human-readable description for the Loadbalancer.
- `tenant_id` - (Optional) Required for admins. The UUID of the tenant who owns the Loadbalancer. Only administrative users can specify a tenant UUID other than their own. Changing this creates a new loadbalancer.
- `vip_address` - (Optional) The ip address of the load balancer. Changing this creates a new loadbalancer.
- `admin_state_up` - (Optional) The administrative state of the Loadbalancer. A valid value is true (UP) or false (DOWN).
- `flavor` - (Optional) The UUID of a flavor. Currently, this is not supported. Changing this creates a new loadbalancer.
- `loadbalancer_provider` - (Optional) The name of the provider. Currently, only vlb is supported. Changing this creates a new loadbalancer.
- `security_group_ids` - (Optional) A list of security group IDs to apply to the loadbalancer. The security groups must be specified by ID and not name (as opposed to how they are configured with the Compute Instance).

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `vip_subnet_id` - See Argument Reference above.
- `name` - See Argument Reference above.

- `description` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `vip_address` - See Argument Reference above.
- `admin_state_up` - See Argument Reference above.
- `flavor` - See Argument Reference above.
- `loadbalancer_provider` - See Argument Reference above.
- `security_group_ids` - See Argument Reference above.
- `vip_port_id` - The Port ID of the Load Balancer IP.

flexibleengine_lb_member_v2

Manages a V2 member resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_lb_member_v2" "member_1" {  
  address      = "192.168.199.23"  
  protocol_port = 8080  
  pool_id      = POOL_ID  
  subnet_id    = SUBNET_ID  
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create an . If omitted, the `region` argument of the provider is used. Changing this creates a new member.
- `pool_id` - (Required) The id of the pool that this member will be assigned to.
- `subnet_id` - (Required) The subnet in which to access the member
- `name` - (Optional) Human-readable name for the member.
- `tenant_id` - (Optional) Required for admins. The UUID of the tenant who owns the member. Only administrative users can specify a tenant UUID other than their own. Changing this creates a new member.
- `address` - (Required) The IP address of the member to receive traffic from the load balancer. Changing this creates a new member.
- `protocol_port` - (Required) The port on which to listen for client traffic. Changing this creates a new member.
- `weight` - (Optional) A positive integer value that indicates the relative portion of traffic that this member should receive from the pool. For example, a member with a weight of 10 receives five times as much traffic as a member with a weight of 2.
- `admin_state_up` - (Optional) The administrative state of the member. A valid value is true (UP) or false (DOWN).

Attributes Reference

The following attributes are exported:

- `id` - The unique ID for the member.
- `name` - See Argument Reference above.
- `weight` - See Argument Reference above.

- `admin_state_up` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `subnet_id` - See Argument Reference above.
- `pool_id` - See Argument Reference above.
- `address` - See Argument Reference above.
- `protocol_port` - See Argument Reference above.

flexibleengine_lb_monitor_v2

Manages a V2 monitor resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_lb_monitor_v2" "monitor_1" {
  pool_id      = "${flexibleengine_lb_pool_v2.pool_1.id}"
  type         = "PING"
  delay        = 20
  timeout      = 10
  max_retries  = 5
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create an . If omitted, the `region` argument of the provider is used. Changing this creates a new monitor.
- `pool_id` - (Required) The id of the pool that this monitor will be assigned to.
- `name` - (Optional) The Name of the Monitor.
- `tenant_id` - (Optional) Required for admins. The UUID of the tenant who owns the monitor. Only administrative users can specify a tenant UUID other than their own. Changing this creates a new monitor.
- `type` - (Required) The type of probe, which is PING, TCP, HTTP, or HTTPS, that is sent by the load balancer to verify the member state. Changing this creates a new monitor.
- `delay` - (Required) The time, in seconds, between sending probes to members.
- `timeout` - (Required) Maximum number of seconds for a monitor to wait for a ping reply before it times out. The value must be less than the delay value.
- `max_retries` - (Required) Number of permissible ping failures before changing the member's status to INACTIVE. Must be a number between 1 and 10..
- `url_path` - (Optional) Required for HTTP(S) types. URI path that will be accessed if monitor type is HTTP or HTTPS.
- `http_method` - (Optional) Required for HTTP(S) types. The HTTP method used for requests by the monitor. If this attribute is not specified, it defaults to "GET".
- `expected_codes` - (Optional) Required for HTTP(S) types. Expected HTTP codes for a passing HTTP(S) monitor. You can either specify a single status like "200", or a range like "200-202".
- `admin_state_up` - (Optional) The administrative state of the monitor. A valid value is true (UP) or false (DOWN).
- `port` - (Optional) Specifies the health check port. The value ranges from 1 to 65536.

Attributes Reference

The following attributes are exported:

- `id` - The unique ID for the monitor.
- `tenant_id` - See Argument Reference above.
- `type` - See Argument Reference above.
- `delay` - See Argument Reference above.
- `timeout` - See Argument Reference above.
- `max_retries` - See Argument Reference above.
- `url_path` - See Argument Reference above.
- `http_method` - See Argument Reference above.
- `expected_codes` - See Argument Reference above.
- `admin_state_up` - See Argument Reference above.
- `port` - See Argument Reference above.

flexibleengine_lb_pool_v2

Manages a V2 pool resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_lb_pool_v2" "pool_1" {
  protocol    = "HTTP"
  lb_method   = "ROUND_ROBIN"
  listener_id = "d9415786-5f1a-428b-b35f-2f1523e146d2"

  persistence {
    type        = "HTTP_COOKIE"
    cookie_name = "testCookie"
  }
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create an . If omitted, the `region` argument of the provider is used. Changing this creates a new pool.
- `tenant_id` - (Optional) Required for admins. The UUID of the tenant who owns the pool. Only administrative users can specify a tenant UUID other than their own. Changing this creates a new pool.
- `name` - (Optional) Human-readable name for the pool.
- `description` - (Optional) Human-readable description for the pool.
- `protocol` = (Required) The protocol - can either be TCP, HTTP or HTTPS. Changing this creates a new pool.
- `loadbalancer_id` - (Optional) The load balancer on which to provision this pool. Changing this creates a new pool. Note: One of `LoadbalancerID` or `ListenerID` must be provided.
- `listener_id` - (Optional) The Listener on which the members of the pool will be associated with. Changing this creates a new pool. Note: One of `LoadbalancerID` or `ListenerID` must be provided.
- `lb_method` - (Required) The load balancing algorithm to distribute traffic to the pool's members. Must be one of `ROUND_ROBIN`, `LEAST_CONNECTIONS`, or `SOURCE_IP`.
- `persistence` - Omit this field to prevent session persistence. Indicates whether connections in the same session will be processed by the same Pool member or not. Changing this creates a new pool.
- `admin_state_up` - (Optional) The administrative state of the pool. A valid value is `true` (UP) or `false` (DOWN).

The `persistence` argument supports:

- `type` - (Required) The type of persistence mode. The current specification supports `SOURCE_IP`, `HTTP_COOKIE`, and `APP_COOKIE`.

- `cookie_name` - (Optional) The name of the cookie if persistence mode is set appropriately. Required if `type = APP_COOKIE`.

Attributes Reference

The following attributes are exported:

- `id` - The unique ID for the pool.
- `tenant_id` - See Argument Reference above.
- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `protocol` - See Argument Reference above.
- `lb_method` - See Argument Reference above.
- `persistence` - See Argument Reference above.
- `admin_state_up` - See Argument Reference above.

flexibleengine_mls_instance_v1

Manages mls instance resource within FlexibleEngine

Example Usage: Creating a MLS instance

```
resource "flexibleengine_mrs_cluster_v1" "cluster1" {
  cluster_name = "mrs-cluster-acc"
  region = "eu-west-0"
  billing_type = 12
  master_node_num = 2
  core_node_num = 3
  master_node_size = "s1.4xlarge.linux.mrs"
  core_node_size = "s1.xlarge.linux.mrs"
  available_zone_id = "eu-west-0a"
  vpc_id = "c1095fe7-03df-4205-ad2d-6f4c181d436e"
  subnet_id = "b65f8d25-c533-47e2-8601-cfaa265a3e3e"
  cluster_version = "MRS 1.3.0"
  volume_type = "SATA"
  volume_size = 100
  safe_mode = 0
  cluster_type = 0
  node_public_cert_name = "KeyPair-ci"
  cluster_admin_secret = ""
  component_list {
    component_name = "Hadoop"
  }
  component_list {
    component_name = "Spark"
  }
  component_list {
    component_name = "Hive"
  }
}

resource "flexibleengine_mls_instance_v1" "instance" {
  name = "terraform-mls-instance"
  version = "1.2.0"
  flavor = "mls.c2.2xlarge.common"
  network {
    vpc_id = "c1095fe7-03df-4205-ad2d-6f4c181d436e"
    subnet_id = "b65f8d25-c533-47e2-8601-cfaa265a3e3e"
    available_zone = "eu-west-0a"
    public_ip {
      bind_type = "not_use"
    }
  }
  mrs_cluster {
    id = "${flexibleengine_mrs_cluster_v1.cluster1.id}"
  }
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the MLS instance. If omitted, the `region` argument of the provider is used. Changing this creates a new instance.
- `name` - (Required) Specifies the MLS instance name. The DB instance name of the same type is unique in the same tenant. Changing this creates a new instance.
- `version` - (Required) Specifies MLS Software version, only `1.2.0` is supported now. Changing this creates a new instance.
- `network` - (Required) Specifies the instance network information. The structure is described below. Changing this creates a new instance.
- `agency` - (Optional) Specifies the agency name. This parameter is mandatory only when you bind an instance to an elastic IP address (EIP). An instance must be bound to an EIP to grant MLS rights to obtain a tenant's token. NOTE: The tenant must create an agency on the Identity and Access Management (IAM) interface in advance. Changing this creates a new instance.
- `flavor` - (Required) Specifies the instance flavor, only `mls.c2.2xlarge.common` is supported now. Changing this creates a new instance.
- `mrs_cluster` - (Required) Specifies the MRS cluster information which the instance is associated. The structure is described below. NOTE: The current MRS instance requires an MRS cluster whose version is 1.3.0 and that is configured with the Spark component. MRS clusters whose version is not 1.3.0 or that are not configured with the Spark component cannot be selected. Changing this creates a new instance.

The `network` block supports:

- `vpc_id` - (Required) Specifies the ID of the virtual private cloud (VPC) where the instance resides. Changing this creates a new instance.
- `subnet_id` - (Required) Specifies the ID of the subnet where the instance resides. Changing this creates a new instance.
- `security_group` - (Optional) Specifies the ID of the security group of the instance. Changing this creates a new instance.
- `available_zone` - (Required) Specifies the AZ of the instance. Changing this creates a new instance.
- `public_ip` - (Required) Specifies the IP address of the instance. The structure is described below. Changing this creates a new instance.

The `public_ip` block supports:

- `bind_type` - (Required) Specifies the bind type. Possible values: `auto_assign` and `not_use`. Changing this creates a new instance.

The `mrs_cluster` block supports:

- `id` - (Required) Specifies the ID of the MRS cluster. Changing this creates a new instance.
- `user_name` - (Optional) Specifies the MRS cluster username. This parameter is mandatory only when the MRS cluster is in the security mode. Changing this creates a new instance.

- `user_password` - (Optional) Specifies the password of the MRS cluster user. The password and username work in a pair. Changing this creates a new instance.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `version` - See Argument Reference above.
- `agency` - See Argument Reference above.
- `flavor` - See Argument Reference above.
- `network/vpc_id` - See Argument Reference above.
- `network/subnet_id` - See Argument Reference above.
- `network/security_group` - See Argument Reference above.
- `network/available_zone` - See Argument Reference above.
- `network/public_ip/bind_type` - See Argument Reference above.
- `network/public_ip/eip_id` - Indicates the EIP ID, This is returned only when `bind_type` is set to `auto_assign`.
- `mrs_cluster` - See Argument Reference above.
- `status` - Indicates the MLS instance status.
- `inner_endpoint` - Indicates the URL for accessing the instance. Only machines in the same VPC and subnet as the instance can access the URL.
- `public_endpoint` - Indicates the URL for accessing the instance. The URL can be accessed from the Internet. The URL is created only after the instance is bound to an EIP.
- `created` - Indicates the creation time in the following format: `yyyy-mm-dd Thh:mm:ssZ`.
- `updated` - Indicates the update time in the following format: `yyyy-mm-dd Thh:mm:ssZ`.

flexibleengine_mrs_cluster_v1

Manages resource cluster within FlexibleEngine MRS.

Example Usage: Creating a MRS cluster

```
resource "flexibleengine_mrs_cluster_v1" "cluster1" {
  cluster_name = "mrs-cluster-chenying"
  region = "sa-chile-1"
  billing_type = 12
  master_node_num = 2
  core_node_num = 3
  master_node_size = "c2.4xlarge.linux.bigdata"
  core_node_size = "s1.xlarge.linux.bigdata"
  available_zone_id = "sa-chile-1a"
  vpc_id = "51edfb75-f9f0-4bbc-b4dc-21466b93f60d"
  subnet_id = "1d7a8646-43ee-455a-a3ab-40da87a1304c"
  cluster_version = "MRS 1.5.0"
  volume_type = "SATA"
  volume_size = 100
  safe_mode = 0
  cluster_type = 0
  node_public_cert_name = "KeyPair-ci"
  cluster_admin_secret = ""
  component_list {
    component_name = "Hadoop"
  }
  component_list {
    component_name = "Spark"
  }
  component_list {
    component_name = "Hive"
  }
}
```

Argument Reference

The following arguments are supported:

- `billing_type` - (Required) The value is 12, indicating on-demand payment.
- `region` - (Required) Cluster region information. Obtain the value from Regions and Endpoints.
- `master_node_num` - (Required) Number of Master nodes The value is 2.
- `master_node_size` - (Required) Best match based on several years of commissioning experience. MRS supports specifications of hosts, and host specifications are determined by CPUs, memory, and disks space. Master nodes support `s1.4xlarge.linux.mrs`, and `s1.8xlarge.linux.mrs`. Core nodes of a streaming cluster support `s1.xlarge.linux.mrs`, `c2.2xlarge.linux.mrs`, `s1.2xlarge.linux.mrs`, `s1.4xlarge.linux.mrs`, `s1.8xlarge.linux.mrs`, and `d1.8xlarge.linux.mrs`. Core nodes of an analysis cluster support all specifications. `s1.xlarge.linux.mrs` CPU: 4-core Memory: 16 GB System Disk: 40

GB c2.2xlarge linux.mrs CPU: 8-core Memory: 16 GB System Disk: 40 GB s1.4xlarge linux.mrs CPU: 16-core Memory: 64 GB System Disk: 40 GB s1.8xlarge linux.mrs CPU: 32-core Memory: 128 GB System Disk: 40 GB d1.xlarge linux.mrs CPU: 6-core Memory: 55 GB System Disk: 40 GB Data Disk: 1.8 TB x 3 HDDs d1.2xlarge linux.mrs CPU: 12-core Memory: 110 GB System Disk: 40 GB Data Disk: 1.8 TB x 6 HDDs d1.4xlarge linux.mrs CPU: 24-core Memory: 220 GB System Disk: 40 GB Data Disk: 1.8 TB x 12 HDDs d1.8xlarge linux.mrs CPU: 48-core Memory: 440 GB System Disk: 40 GB Data Disk: 1.8 TB x 24 HDDs

- `core_node_num` - (Required) Number of Core nodes Value range: 3 to 100 A maximum of 100 Core nodes are supported by default. If more than 100 Core nodes are required, contact technical support engineers or invoke background APIs to modify the database.
- `core_node_size` - (Required) Instance specification of a Core node Configuration method of this parameter is identical to that of `master_node_size`.
- `available_zone_id` - (Required) ID of an available zone. Obtain the value from Regions and Endpoints.
- `cluster_name` - (Required) Cluster name, which is globally unique and contains only 1 to 64 letters, digits, hyphens (-), and underscores (_).
- `vpc_id` - (Required) ID of the VPC where the subnet locates Obtain the VPC ID from the management console as follows: Register an account and log in to the management console. Click Virtual Private Cloud and select Virtual Private Cloud from the left list. On the Virtual Private Cloud page, obtain the VPC ID from the list.
- `subnet_id` - (Required) Subnet ID Obtain the subnet ID from the management console as follows: Register an account and log in to the management console. Click Virtual Private Cloud and select Virtual Private Cloud from the left list. On the Virtual Private Cloud page, obtain the subnet ID from the list.
- `cluster_version` - (Optional) Version of the clusters Currently, MRS 1.3.0 and MRS 1.5.0 are supported. The latest version of MRS is used by default. Currently, the latest version is MRS 1.5.0.
- `cluster_type` - (Optional) Type of clusters 0: analysis cluster 1: streaming cluster The default value is 0.
- `volume_type` - (Required) Type of disks SATA and SSD are supported. SATA: common I/O SSD: super high-speed I/O
- `volume_size` - (Required) Data disk storage space of a Core node Users can add disks to expand storage capacity when creating a cluster. There are the following scenarios: Separation of data storage and computing: Data is stored in the OBS system. Costs of clusters are relatively low but computing performance is poor. The clusters can be deleted at any time. It is recommended when data computing is not frequently performed. Integration of data storage and computing: Data is stored in the HDFS system. Costs of clusters are relatively high but computing performance is good. The clusters cannot be deleted in a short term. It is recommended when data computing is frequently performed. Value range: 100 GB to 32000 GB
- `node_public_cert_name` - (Required) Name of a key pair You can use a key to log in to the Master node in the cluster.
- `safe_mode` - (Required) MRS cluster running mode 0: common mode The value indicates that the Kerberos authentication is disabled. Users can use all functions provided by the cluster. 1: safe mode The value indicates that the Kerberos authentication is enabled. Common users cannot use the file management or job management functions of an MRS cluster and cannot view cluster resource usage or the job records of Hadoop and Spark. To use these functions, the users must obtain the relevant permissions from the MRS Manager administrator. The request has the `cluster_admin_secret` parameter only when `safe_mode` is set to 1.

- `cluster_admin_secret` - (Optional) Indicates the password of the MRS Manager administrator. The password for MRS 1.5.0: Must contain 6 to 32 characters. Must contain at least two types of the following: Lowercase letters Uppercase letters Digits Special characters of `~!@#%&^×*()-_+=\|[{ }];:','<.>/?` Spaces Must be different from the username. Must be different from the username written in reverse order. The password for MRS 1.3.0: Must contain 8 to 64 characters. Must contain at least four types of the following: Lowercase letters Uppercase letters Digits Special characters of `~!@#%&*()_+=|[{}];:','<.>/?` Spaces Must be different from the username. Must be different from the username written in reverse order. This parameter needs to be configured only when `safe_mode` is set to 1.
- `log_collection` - (Optional) Indicates whether logs are collected when cluster installation fails. 0: not collected 1: collected The default value is 0. If `log_collection` is set to 1, OBS buckets will be created to collect the MRS logs. These buckets will be charged.
- `component_list` - (Required) Service component list.
- `add_jobs` - (Optional) You can submit a job when you create a cluster to save time and use MRS easily. Only one job can be added.

The `component_list` block supports:

- `component_name` - (Required) Component name Currently, Hadoop, Spark, HBase, Hive, Hue, Loader, Flume, Kafka and Storm are supported. Loader and Flume are not supported by MRS 1.3.0.
- `componentId` - Component ID Component IDs supported by MRS 1.5.0 include: MRS 1.5.0_001: Hadoop MRS 1.5.0_002: Spark MRS 1.5.0_003: HBase MRS 1.5.0_004: Hive MRS 1.5.0_005: Hue MRS 1.5.0_006: Kafka MRS 1.5.0_007: Storm MRS 1.5.0_008: Loader MRS 1.5.0_009: Flume Component IDs supported by MRS 1.3.0 include: MRS 1.3.0_001: Hadoop MRS 1.3.0_002: Spark MRS 1.3.0_003: HBase MRS 1.3.0_004: Hive MRS 1.3.0_005: Hue MRS 1.3.0_006: Kafka MRS 1.3.0_007: Storm For example, the component ID of Hadoop is MRS 1.5.0_001, or MRS 1.3.0_001.
- `componentName` - Component name Currently, Hadoop, Spark, HBase, Hive, Hue, Loader, Flume, Kafka and Storm are supported. Loader and Flume are not supported by MRS 1.3.0.
- `componentVersion` - Component version MRS 1.5.0 supports the following component version: Component version of an analysis cluster: Hadoop: 2.7.2 Spark: 2.1.0 HBase: 1.0.2 Hive: 1.2.1 Hue: 3.11.0 Loader: 2.0.0 Component version of a streaming cluster: Kafka: 0.10.0.0 Storm: 1.0.2 Flume: 1.6.0 MRS 1.3.0 supports the following component version: Component version of an analysis cluster: Hadoop: 2.7.2 Spark: 1.5.1 HBase: 1.0.2 Hive: 1.2.1 Hue: 3.11.0 Component version of a streaming cluster: Kafka: 0.10.0.0 Storm: 1.0.2
- `componentDesc` - Component description

The `add_jobs` block supports: * `job_type` - (Required) Job type 1: MapReduce 2: Spark 3: Hive Script 4: HiveQL (not supported currently) 5: DistCp, importing and exporting data (not supported in this API currently). 6: Spark Script 7: Spark SQL, submitting Spark SQL statements (not supported in this API currently). NOTE: Spark and Hive jobs can be added to only clusters including Spark and Hive components.

- `job_name` - (Required) Job name It contains only 1 to 64 letters, digits, hyphens (-), and underscores (_). NOTE: Identical job names are allowed but not recommended.
- `jar_path` - (Required) Path of the .jar file or .sql file for program execution The parameter must meet the following requirements: Contains a maximum of 1023 characters, excluding special characters such as `;&|><`$`. The address cannot be empty or full of spaces. Starts with `/` or `s3a://`. Spark Script must end with `.sql`; while MapReduce and Spark Jar must end with `.jar`. `sql` and `jar` are case-insensitive.

- `arguments` - (Optional) Key parameter for program execution The parameter is specified by the function of the user's program. MRS is only responsible for loading the parameter. The parameter contains a maximum of 2047 characters, excluding special characters such as `;&>'<$`, and can be empty.
- `input` - (Optional) Path for inputting data, which must start with `/` or `s3a://`. A correct OBS path is required. The parameter contains a maximum of 1023 characters, excluding special characters such as `;&>'<$`, and can be empty.
- `output` - (Optional) Path for outputting data, which must start with `/` or `s3a://`. A correct OBS path is required. If the path does not exist, the system automatically creates it. The parameter contains a maximum of 1023 characters, excluding special characters such as `;&>'<$`, and can be empty.
- `job_log` - (Optional) Path for storing job logs that record job running status. This path must start with `/` or `s3a://`. A correct OBS path is required. The parameter contains a maximum of 1023 characters, excluding special characters such as `;&>'<$`, and can be empty.
- `shutdown_cluster` - (Optional) Whether to delete the cluster after the jobs are complete true: Yes false: No
- `file_action` - (Optional) Data import and export import export
- `submit_job_once_cluster_run` - (Required) true: A job is submitted when a cluster is created. false: A job is submitted separately. The parameter is set to true in this example.
- `hql` - (Optional) HiveQL statement
- `hive_script_path` - (Optional) SQL program path This parameter is needed by Spark Script and Hive Script jobs only and must meet the following requirements: Contains a maximum of 1023 characters, excluding special characters such as `;&>'<$`. The address cannot be empty or full of spaces. Starts with `/` or `s3a://`. Ends with `.sql`. sql is case-insensitive.

Attributes Reference

The following attributes are exported:

- `billing_type` - See Argument Reference above.
- `data_center` - See Argument Reference above.
- `master_node_num` - See Argument Reference above.
- `master_node_size` - See Argument Reference above.
- `core_node_num` - See Argument Reference above.
- `core_node_size` - See Argument Reference above.
- `available_zone_id` - See Argument Reference above.
- `cluster_name` - See Argument Reference above.
- `vpc_id` - See Argument Reference above.
- `subnet_id` - See Argument Reference above.
- `cluster_version` - See Argument Reference above.
- `cluster_type` - See Argument Reference above.

- volume_type - See Argument Reference above.
- volume_size - See Argument Reference above.
- node_public_cert_name - See Argument Reference above.
- safe_mode - See Argument Reference above.
- cluster_admin_secret - See Argument Reference above.
- log_collection - See Argument Reference above.
- component_list - See Argument Reference below.
- add_jobs - See Argument Reference above.
- order_id - Order ID for creating clusters.
- cluster_id - Cluster ID.
- available_zone_name - Name of an availability zone.
- instance_id - Instance ID.
- hadoop_version - Hadoop version.
- master_node_ip - IP address of a Master node.
- externalIp - Internal IP address.
- private_ip_first - Primary private IP address.
- external_ip - External IP address.
- slave_security_groups_id - Standby security group ID.
- security_groups_id - Security group ID.
- external_alternate_ip - Backup external IP address.
- master_node_spec_id - Specification ID of a Master node.
- core_node_spec_id - Specification ID of a Core node.
- master_node_product_id - Product ID of a Master node.
- core_node_product_id - Product ID of a Core node.
- duration - Cluster subscription duration.
- vnc - URI address for remote login of the elastic cloud server.
- fee - Cluster creation fee, which is automatically calculated.
- deployment_id - Deployment ID of a cluster.
- cluster_state - Cluster status Valid values include: existing history starting running terminated failed abnormal terminating rebooting shutdown frozen scaling-out scaling-in scaling-error.
- tenant_id - Project ID.

- `create_at` - Cluster creation time.
- `update_at` - Cluster update time.
- `error_info` - Error information.
- `charging_start_time` - Time when charging starts.
- `remark` - Remarks of a cluster.

The `component_list` attributes:

- * `component_name` - (Required) Component name Currently, Hadoop, Spark, HBase, Hive, Hue, Loader, Flume, Kafka and Storm are supported. Loader and Flume are not supported by MRS 1.3.0.
- * `component_id` - Component ID Component IDs supported by MRS 1.5.0 include: MRS 1.5.0_001: Hadoop MRS 1.5.0_002: Spark MRS 1.5.0_003: HBase MRS 1.5.0_004: Hive MRS 1.5.0_005: Hue MRS 1.5.0_006: Kafka MRS 1.5.0_007: Storm MRS 1.5.0_008: Loader MRS 1.5.0_009: Flume Component IDs supported by MRS 1.3.0 include: MRS 1.3.0_001: Hadoop MRS 1.3.0_002: Spark MRS 1.3.0_003: HBase MRS 1.3.0_004: Hive MRS 1.3.0_005: Hue MRS 1.3.0_006: Kafka MRS 1.3.0_007: Storm For example, the component ID of Hadoop is MRS 1.5.0_001, or MRS 1.3.0_001.
- * `component_version` - Component version MRS 1.5.0 supports the following component version: Component version of an analysis cluster: Hadoop: 2.7.2 Spark: 2.1.0 HBase: 1.0.2 Hive: 1.2.1 Hue: 3.11.0 Loader: 2.0.0 Component version of a streaming cluster: Kafka: 0.10.0.0 Storm: 1.0.2 Flume: 1.6.0 MRS 1.3.0 supports the following component version: Component version of an analysis cluster: Hadoop: 2.7.2 Spark: 1.5.1 HBase: 1.0.2 Hive: 1.2.1 Hue: 3.11.0 Component version of a streaming cluster: Kafka: 0.10.0.0 Storm: 1.0.2
- * `component_desc` - Component description

flexibleengine_mrs_job_v1

Manages resource job within FlexibleEngine MRS.

Example Usage

```
resource "flexibleengine_mrs_job_v1" "job1" {
  job_type = 1
  job_name = "test_mapreduce_job1"
  cluster_id = "ef43d2ff-1ecf-4f13-bd0c-0004c429a058"
  jar_path = "s3a://wordcount/program/hadoop-mapreduce-examples-2.7.5.jar"
  input = "s3a://wordcount/input/"
  output = "s3a://wordcount/output/"
  job_log = "s3a://wordcount/log/"
  arguments = "wordcount"
}
```

Argument Reference

The following arguments are supported:

- `job_type` - (Required) Job type 1: MapReduce 2: Spark 3: Hive Script 4: HiveQL (not supported currently) 5: DistCp, importing and exporting data. 6: Spark Script 7: Spark SQL, submitting Spark SQL statements. (not supported in this API currently) NOTE: Spark and Hive jobs can be added to only clusters including Spark and Hive components.
- `job_name` - (Required) Job name Contains only 1 to 64 letters, digits, hyphens (-), and underscores (_). NOTE: Identical job names are allowed but not recommended.
- `cluster_id` - (Required) Cluster ID
- `jar_path` - (Required) Path of the .jar package or .sql file for program execution The parameter must meet the following requirements: Contains a maximum of 1023 characters, excluding special characters such as ; | & > < '\$. The address cannot be empty or full of spaces. Starts with / or s3a://. Spark Script must end with .sql; while MapReduce and Spark Jar must end with .jar. sql and jar are case-insensitive.
- `arguments` - (Optional) Key parameter for program execution. The parameter is specified by the function of the user's program. MRS is only responsible for loading the parameter. The parameter contains a maximum of 2047 characters, excluding special characters such as ; | & > < '\$, and can be empty.
- `input` - (Optional) Path for inputting data, which must start with / or s3a://. A correct OBS path is required. The parameter contains a maximum of 1023 characters, excluding special characters such as ; | & > < '\$, and can be empty.
- `output` - (Optional) Path for outputting data, which must start with / or s3a://. A correct OBS path is required. If the path does not exist, the system automatically creates it. The parameter contains a maximum of 1023 characters, excluding special characters such as ; | & > < '\$, and can be empty.

- `job_log` - (Optional) Path for storing job logs that record job running status. This path must start with `/` or `s3a://`. A correct OBS path is required. The parameter contains a maximum of 1023 characters, excluding special characters such as `;>'<$`, and can be empty.
- `hive_script_path` - (Optional) SQL program path This parameter is needed by Spark Script and Hive Script jobs only and must meet the following requirements: Contains a maximum of 1023 characters, excluding special characters such as `;>'<$`. The address cannot be empty or full of spaces. Starts with `/` or `s3a://`. Ends with `.sql`. `sql` is case-insensitive.
- `is_protected` - (Optional) Whether a job is protected true false The current version does not support this function.
- `is_public` - (Optional) Whether a job is public true false The current version does not support this function.

Attributes Reference

The following attributes are exported:

- `job_type` - See Argument Reference above.
- `job_name` - See Argument Reference above.
- `cluster_id` - See Argument Reference above.
- `jar_path` - See Argument Reference above.
- `arguments` - See Argument Reference above.
- `input` - See Argument Reference above.
- `output` - See Argument Reference above.
- `job_log` - See Argument Reference above.
- `hive_script_path` - See Argument Reference above.
- `is_protected` - See Argument Reference above.
- `is_public` - See Argument Reference above.

flexibleengine_nat_dnat_rule_v2

Example Usage

Dnat

```
resource "flexibleengine_nat_dnat_rule_v2" "dnat_1" {
  floating_ip_id = "2bd659ab-bbf7-43d7-928b-9ee6a10de3ef"
  nat_gateway_id = "bf99c679-9f41-4dac-8513-9c9228e713e1"
  private_ip = "10.0.0.12"
  internal_service_port = 993
  protocol = "tcp"
  external_service_port = 242
}
```

Argument Reference

The following arguments are supported:

- `floating_ip_id` - (Required) Specifies the ID of the floating IP address. Changing this creates a new resource.
- `internal_service_port` - (Required) Specifies port used by ECSs or BMSs to provide services for external systems. Changing this creates a new resource.
- `nat_gateway_id` - (Required) ID of the nat gateway this dnat rule belongs to. Changing this creates a new dnat rule.
- `port_id` - (Optional) Specifies the port ID of an ECS or a BMS. This parameter and `private_ip` are alternative. Changing this creates a new dnat rule.
- `private_ip` - (Optional) Specifies the private IP address of a user, for example, the IP address of a VPC for dedicated connection. This parameter and `port_id` are alternative. Changing this creates a new dnat rule.
- `protocol` - (Required) Specifies the protocol type. Currently, TCP, UDP, and ANY are supported. The protocol number of TCP, UDP, and ANY is 6, 17, and 0, respectively. Changing this creates a new dnat rule.
- `external_service_port` - (Required) Specifies port used by ECSs or BMSs to provide services for external systems. Changing this creates a new dnat rule.

Attributes Reference

In addition to the arguments listed above, the following computed attributes are exported:

- `created_at` - Dnat rule creation time.
- `status` - Dnat rule status.

Import

Dnat can be imported using the following format:

```
$ terraform import flexibleengine_nat_dnat_rule_v2.dnat_1 f4f783a7-b908-4215-b018-724960e5df4a
```

flexibleengine_nat_gateway_v2

Manages a V2 nat gateway resource within FlexibleEngine Nat

Example Usage

```
resource "flexibleengine_nat_gateway_v2" "nat_1" {
  name      = "Terraform"
  description = "test for terraform2"
  spec      = "3"
  router_id = "2c1fe4bd-ebad-44ca-ae9d-e94e63847b75"
  internal_network_id = "dc8632e2-d9ff-41b1-aa0c-d455557314a0"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 nat client. If omitted, the `region` argument of the provider is used. Changing this creates a new nat gateway.
- `name` - (Required) The name of the nat gateway.
- `description` - (Optional) The description of the nat gateway.
- `spec` - (Required) The specification of the nat gateway, valid values are "1", "2", "3", "4".
- `tenant_id` - (Optional) The target tenant ID in which to allocate the nat gateway. Changing this creates a new nat gateway.
- `router_id` - (Required) ID of the router this nat gateway belongs to. Changing this creates a new nat gateway.
- `internal_network_id` - (Required) ID of the network this nat gateway connects to. Changing this creates a new nat gateway.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `spec` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.

- `router_id` - See Argument Reference above.
- `internal_network_id` - See Argument Reference above.

flexibleengine_nat_snat_rule_v2

Manages a V2 snat rule resource within FlexibleEngine Nat

Example Usage

```
resource "flexibleengine_nat_snat_rule_v2" "snat_1" {  
  nat_gateway_id = "3c0dffda-7c76-452b-9dcc-5bce7ae56b17"  
  network_id = "dc8632e2-d9ff-41b1-aa0c-d455557314a0"  
  floating_ip_id = "0a166fc5-a904-42fb-b1ef-cf18afeeddca"  
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 nat client. If omitted, the `region` argument of the provider is used. Changing this creates a new snat rule.
- `nat_gateway_id` - (Required) ID of the nat gateway this snat rule belongs to. Changing this creates a new snat rule.
- `network_id` - (Required) ID of the network this snat rule connects to. Changing this creates a new snat rule.
- `floating_ip_id` - (Required) ID of the floating ip this snat rule connects to. Changing this creates a new snat rule.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `nat_gateway_id` - See Argument Reference above.
- `network_id` - See Argument Reference above.
- `floating_ip_id` - See Argument Reference above.

flexibleengine_networking_floatingip_associate_v2

Associates a floating IP to a port. This is useful for situations where you have a pre-allocated floating IP or are unable to use the `flexibleengine_networking_floatingip_v2` resource to create a floating IP.

Example Usage

```
resource "flexibleengine_networking_port_v2" "port_1" {
  network_id = "a5bbd213-e1d3-49b6-aed1-9df60ea94b9a"
}

resource "flexibleengine_networking_floatingip_associate_v2" "fip_1" {
  floating_ip = "1.2.3.4"
  port_id = "${flexibleengine_networking_port_v2.port_1.id}"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create a floating IP that can be used with another networking resource, such as a load balancer. If omitted, the `region` argument of the provider is used. Changing this creates a new floating IP (which may or may not have a different address).
- `floating_ip` - (Required) IP Address of an existing floating IP.
- `port_id` - (Required) ID of an existing port with at least one IP address to associate with this floating IP.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `floating_ip` - See Argument Reference above.
- `port_id` - See Argument Reference above.

Import

Floating IP associations can be imported using the `id` of the floating IP, e.g.

```
$ terraform import flexibleengine_networking_floatingip_associate_v2.fip 2c7f39f3-702b-48d1-940c-b50384177ee1
```

flexibleengine_networking_floatingip_v2

Manages a V2 floating IP resource within FlexibleEngine Neutron (networking) that can be used for load balancers. These are similar to Nova (compute) floating IP resources, but only compute floating IPs can be used with compute instances.

Example Usage

```
resource "flexibleengine_networking_floatingip_v2" "floatip_1" {  
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create a floating IP that can be used with another networking resource, such as a load balancer. If omitted, the `region` argument of the provider is used. Changing this creates a new floating IP (which may or may not have a different address).
- `pool` - (Optional) The name of the pool from which to obtain the floating IP. Default value is `admin_external_net`. Changing this creates a new floating IP.
- `port_id` - (Optional) ID of an existing port with at least one IP address to associate with this floating IP.
- `tenant_id` - (Optional) The target tenant ID in which to allocate the floating IP, if you specify this together with a `port_id`, make sure the target port belongs to the same tenant. Changing this creates a new floating IP (which may or may not have a different address)
- `fixed_ip` - Fixed IP of the port to associate with this floating IP. Required if the port has multiple fixed IPs.
- `value_specs` - (Optional) Map of additional options.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `pool` - See Argument Reference above.
- `address` - The actual floating IP address itself.
- `port_id` - ID of associated port.
- `tenant_id` - the ID of the tenant in which to create the floating IP.
- `fixed_ip` - The fixed IP which the floating IP maps to.

Import

Floating IPs can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_networking_floatingip_v2.floatip_1 2c7f39f3-702b-48d1-940c-b50384177ee1
```

flexibleengine_networking_network_v2

Manages a V2 Neutron network resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_networking_network_v2" "network_1" {
  name          = "network_1"
  admin_state_up = "true"
}

resource "flexibleengine_networking_subnet_v2" "subnet_1" {
  name          = "subnet_1"
  network_id    = "${flexibleengine_networking_network_v2.network_1.id}"
  cidr          = "192.168.199.0/24"
  ip_version    = 4
}

resource "flexibleengine_compute_secgroup_v2" "secgroup_1" {
  name          = "secgroup_1"
  description   = "a security group"

  rule {
    from_port = 22
    to_port   = 22
    ip_protocol = "tcp"
    cidr      = "0.0.0.0/0"
  }
}

resource "flexibleengine_networking_port_v2" "port_1" {
  name          = "port_1"
  network_id    = "${flexibleengine_networking_network_v2.network_1.id}"
  admin_state_up = "true"
  security_group_ids = ["${flexibleengine_compute_secgroup_v2.secgroup_1.id}"]

  fixed_ip {
    "subnet_id" = "${flexibleengine_networking_subnet_v2.subnet_1.id}"
    "ip_address" = "192.168.199.10"
  }
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name          = "instance_1"
  security_groups = ["${flexibleengine_compute_secgroup_v2.secgroup_1.name}"]

  network {
    port = "${flexibleengine_networking_port_v2.port_1.id}"
  }
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create a Neutron network. If omitted, the `region` argument of the provider is used. Changing this creates a new network.
- `name` - (Optional) The name of the network. Changing this updates the name of the existing network.
- `shared` - (Optional) Specifies whether the network resource can be accessed by any tenant or not. Changing this updates the sharing capabilities of the existing network.
- `tenant_id` - (Optional) The owner of the network. Required if admin wants to create a network for another tenant. Changing this creates a new network.
- `admin_state_up` - (Optional) The administrative state of the network. Acceptable values are "true" and "false". Changing this value updates the state of the existing network.
- `segments` - (Optional) An array of one or more provider segment objects.
- `value_specs` - (Optional) Map of additional options.

The `segments` block supports:

- `physical_network` - The physical network where this network is implemented.
- `segmentation_id` - An isolated segment on the physical network.
- `network_type` - The type of physical network.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `shared` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `admin_state_up` - See Argument Reference above.

Import

Networks can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_networking_network_v2.network_1 d90ce693-5ccf-4136-a0ed-152ce412b6b9
```

flexibleengine_networking_port_v2

Manages a V2 port resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_networking_network_v2" "network_1" {
  name          = "network_1"
  admin_state_up = "true"
}

resource "flexibleengine_networking_port_v2" "port_1" {
  name          = "port_1"
  network_id    = "${flexibleengine_networking_network_v2.network_1.id}"
  admin_state_up = "true"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 networking client. A networking client is needed to create a port. If omitted, the `region` argument of the provider is used. Changing this creates a new port.
- `name` - (Optional) A unique name for the port. Changing this updates the `name` of an existing port.
- `network_id` - (Required) The ID of the network to attach the port to. Changing this creates a new port.
- `admin_state_up` - (Optional) Administrative up/down status for the port (must be "true" or "false" if provided). Changing this updates the `admin_state_up` of an existing port.
- `mac_address` - (Optional) Specify a specific MAC address for the port. Changing this creates a new port.
- `tenant_id` - (Optional) The owner of the Port. Required if admin wants to create a port for another tenant. Changing this creates a new port.
- `device_owner` - (Optional) The device owner of the Port. Changing this creates a new port.
- `security_group_ids` - (Optional) A list of security group IDs to apply to the port. The security groups must be specified by ID and not name (as opposed to how they are configured with the Compute Instance).
- `device_id` - (Optional) The ID of the device attached to the port. Changing this creates a new port.
- `fixed_ip` - (Optional) An array of desired IPs for this port. The structure is described below.
- `allowed_address_pairs` - (Optional) An IP/MAC Address pair of additional IP addresses that can be active on this port. The structure is described below.
- `value_specs` - (Optional) Map of additional options.

The `fixed_ip` block supports:

- `subnet_id` - (Required) Subnet in which to allocate IP address for this port.
- `ip_address` - (Optional) IP address desired in the subnet for this port. If you don't specify `ip_address`, an available IP address from the specified subnet will be allocated to this port.

The `allowed_address_pairs` block supports:

- `ip_address` - (Required) The additional IP address.
- `mac_address` - (Optional) The additional MAC address.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `admin_state_up` - See Argument Reference above.
- `mac_address` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `device_owner` - See Argument Reference above.
- `security_group_ids` - See Argument Reference above.
- `device_id` - See Argument Reference above.
- `fixed_ip` - See Argument Reference above.
- `all_fixed_ips` - The collection of Fixed IP addresses on the port in the order returned by the Network v2 API.

Import

Ports can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_networking_port_v2.port_1 eae26a3e-1c33-4cc1-9c31-0cd729c438a1
```

Notes

Ports and Instances

There are some notes to consider when connecting Instances to networks using Ports. Please see the `flexibleengine_compute_instance_v2` documentation for further documentation.

flexibleengine_networking_router_interface_v2

Manages a V2 router interface resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_networking_network_v2" "network_1" {
  name          = "tf_test_network"
  admin_state_up = "true"
}

resource "flexibleengine_networking_subnet_v2" "subnet_1" {
  network_id = "${flexibleengine_networking_network_v2.network_1.id}"
  cidr       = "192.168.199.0/24"
  ip_version = 4
}

resource "flexibleengine_networking_router_v2" "router_1" {
  name          = "my_router"
  external_gateway = "f67f0d72-0ddf-11e4-9d95-e1f29f417e2f"
}

resource "flexibleengine_networking_router_interface_v2" "router_interface_1" {
  router_id = "${flexibleengine_networking_router_v2.router_1.id}"
  subnet_id = "${flexibleengine_networking_subnet_v2.subnet_1.id}"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 networking client. A networking client is needed to create a router. If omitted, the `region` argument of the provider is used. Changing this creates a new router interface.
- `router_id` - (Required) ID of the router this interface belongs to. Changing this creates a new router interface.
- `subnet_id` - ID of the subnet this interface connects to. Changing this creates a new router interface.
- `port_id` - ID of the port this interface connects to. Changing this creates a new router interface.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `router_id` - See Argument Reference above.
- `subnet_id` - See Argument Reference above.

- `port_id` - See Argument Reference above.

flexibleengine_networking_router_route_v2

Creates a routing entry on a FlexibleEngine V2 router.

Example Usage

```
resource "flexibleengine_networking_router_v2" "router_1" {
  name          = "router_1"
  admin_state_up = "true"
}

resource "flexibleengine_networking_network_v2" "network_1" {
  name          = "network_1"
  admin_state_up = "true"
}

resource "flexibleengine_networking_subnet_v2" "subnet_1" {
  network_id = "${flexibleengine_networking_network_v2.network_1.id}"
  cidr       = "192.168.199.0/24"
  ip_version = 4
}

resource "flexibleengine_networking_router_interface_v2" "int_1" {
  router_id = "${flexibleengine_networking_router_v2.router_1.id}"
  subnet_id = "${flexibleengine_networking_subnet_v2.subnet_1.id}"
}

resource "flexibleengine_networking_router_route_v2" "router_route_1" {
  depends_on = ["flexibleengine_networking_router_interface_v2.int_1"]
  router_id   = "${flexibleengine_networking_router_v2.router_1.id}"
  destination_cidr = "10.0.1.0/24"
  next_hop     = "192.168.199.254"
}
```

Argument Reference

The following arguments are supported:

- **region** - (Optional) The region in which to obtain the V2 networking client. A networking client is needed to configure a routing entry on a router. If omitted, the `region` argument of the provider is used. Changing this creates a new routing entry.
- **router_id** - (Required) ID of the router this routing entry belongs to. Changing this creates a new routing entry.
- **destination_cidr** - (Required) CIDR block to match on the packet's destination IP. Changing this creates a new routing entry.
- **next_hop** - (Required) IP address of the next hop gateway. Changing this creates a new routing entry.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `router_id` - See Argument Reference above.
- `destination_cidr` - See Argument Reference above.
- `next_hop` - See Argument Reference above.

Notes

The `next_hop` IP address must be directly reachable from the router at the `flexibleengine_networking_router_route_v2` resource creation time. You can ensure that by explicitly specifying a dependency on the `flexibleengine_networking_router_interface_v2` resource that connects the next hop to the router, as in the example above.

flexibleengine_networking_router_v2

Manages a V2 router resource within FlexibleEngine. The router is the top-level resource for the VPC within FlexibleEngine.

Example Usage

```
resource "flexibleengine_networking_router_v2" "router_1" {
  name          = "my_router"
  external_gateway = "f67f0d72-0ddf-11e4-9d95-e1f29f417e2f"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 networking client. A networking client is needed to create a router. If omitted, the `region` argument of the provider is used. Changing this creates a new router.
- `name` - (Optional) A unique name for the router. Changing this updates the `name` of an existing router.
- `admin_state_up` - (Optional) Administrative up/down status for the router (must be "true" or "false" if provided). Changing this updates the `admin_state_up` of an existing router.
- `distributed` - (Optional) Indicates whether or not to create a distributed router. The default policy setting in Neutron restricts usage of this property to administrative users only.
- `external_gateway` - (Optional) The network UUID of an external gateway for the router. A router with an external gateway is required if any compute instances or load balancers will be using floating IPs. Changing this updates the `external_gateway` of an existing router.
- `enable_snat` - (Optional) Enable Source NAT for the router. Valid values are "true" or "false". An `external_gateway` has to be set in order to set this property. Changing this updates the `enable_snat` of the router.
- `tenant_id` - (Optional) The owner of the floating IP. Required if admin wants to create a router for another tenant. Changing this creates a new router.
- `value_specs` - (Optional) Map of additional driver-specific options.

Attributes Reference

The following attributes are exported:

- `id` - ID of the router.
- `region` - See Argument Reference above.
- `name` - See Argument Reference above.

- `admin_state_up` - See Argument Reference above.
- `external_gateway` - See Argument Reference above.
- `enable_snat` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `value_specs` - See Argument Reference above.

flexibleengine_networking_secgroup_rule_v2

Manages a V2 neutron security group rule resource within FlexibleEngine. Unlike Nova security groups, neutron separates the group from the rules and also allows an admin to target a specific tenant_id.

Example Usage

```
resource "flexibleengine_networking_secgroup_v2" "secgroup_1" {
  name          = "secgroup_1"
  description   = "My neutron security group"
}

resource "flexibleengine_networking_secgroup_rule_v2" "secgroup_rule_1" {
  direction      = "ingress"
  ethertype      = "IPv4"
  protocol       = "tcp"
  port_range_min = 22
  port_range_max = 22
  remote_ip_prefix = "0.0.0.0/0"
  security_group_id = "${flexibleengine_networking_secgroup_v2.secgroup_1.id}"
}
```

Argument Reference

The following arguments are supported:

- **region** - (Optional) The region in which to obtain the V2 networking client. A networking client is needed to create a port. If omitted, the `region` argument of the provider is used. Changing this creates a new security group rule.
- **direction** - (Required) The direction of the rule, valid values are **ingress** or **egress**. Changing this creates a new security group rule.
- **ethertype** - (Required) The layer 3 protocol type, valid values are **IPv4** or **IPv6**. Changing this creates a new security group rule.
- **protocol** - (Optional) The layer 4 protocol type, valid values are following. Changing this creates a new security group rule. This is required if you want to specify a port range.
 - **tcp**
 - **udp**
 - **icmp**
 - **ah**
 - **dccp**
 - **egp**
 - **esp**

- gre
 - igmp
 - ipv6-encap
 - ipv6-frag
 - ipv6-icmp
 - ipv6-nonxt
 - ipv6-opts
 - ipv6-route
 - ospf
 - pgm
 - rsvp
 - sctp
 - udplite
 - vrrp
- port_range_min - (Optional) The lower part of the allowed port range, valid integer value needs to be between 1 and 65535. Changing this creates a new security group rule.
 - port_range_max - (Optional) The higher part of the allowed port range, valid integer value needs to be between 1 and 65535. Changing this creates a new security group rule.
 - remote_ip_prefix - (Optional) The remote CIDR, the value needs to be a valid CIDR (i.e. 192.168.0.0/16). Changing this creates a new security group rule.
 - remote_group_id - (Optional) The remote group id, the value needs to be an FlexibleEngine ID of a security group in the same tenant. Changing this creates a new security group rule.
 - security_group_id - (Required) The security group id the rule should belong to, the value needs to be an FlexibleEngine ID of a security group in the same tenant. Changing this creates a new security group rule.
 - tenant_id - (Optional) The owner of the security group. Required if admin wants to create a port for another tenant. Changing this creates a new security group rule.

Attributes Reference

The following attributes are exported:

- region - See Argument Reference above.
- direction - See Argument Reference above.
- ethertype - See Argument Reference above.
- protocol - See Argument Reference above.

- `port_range_min` - See Argument Reference above.
- `port_range_max` - See Argument Reference above.
- `remote_ip_prefix` - See Argument Reference above.
- `remote_group_id` - See Argument Reference above.
- `security_group_id` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.

Import

Security Group Rules can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_networking_secgroup_rule_v2.secgroup_rule_1 aeb68ee3-6e9d-4256-955c-9584a6212745
```

flexibleengine_networking_secgroup_v2

Manages a V2 neutron security group resource within FlexibleEngine. Unlike Nova security groups, neutron separates the group from the rules and also allows an admin to target a specific tenant_id.

Example Usage

```
resource "flexibleengine_networking_secgroup_v2" "secgroup_1" {
  name          = "secgroup_1"
  description   = "My neutron security group"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 networking client. A networking client is needed to create a port. If omitted, the `region` argument of the provider is used. Changing this creates a new security group.
- `name` - (Required) A unique name for the security group.
- `description` - (Optional) A unique name for the security group.
- `tenant_id` - (Optional) The owner of the security group. Required if admin wants to create a port for another tenant. Changing this creates a new security group.
- `delete_default_rules` - (Optional) Whether or not to delete the default egress security rules. This is `false` by default. See the below note for more information.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `description` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.

Default Security Group Rules

In most cases, FlexibleEngine will create some egress security group rules for each new security group. These security group rules will not be managed by Terraform, so if you prefer to have *all* aspects of your infrastructure managed by Terraform, set `delete_default_rules` to `true` and then create separate security group rules such as the following:

```
resource "flexibleengine_networking_secgroup_rule_v2" "secgroup_rule_v4" {
  direction = "egress"
  ethertype = "IPv4"
  security_group_id = "${flexibleengine_networking_secgroup_v2.secgroup.id}"
}

resource "flexibleengine_networking_secgroup_rule_v2" "secgroup_rule_v6" {
  direction = "egress"
  ethertype = "IPv6"
  security_group_id = "${flexibleengine_networking_secgroup_v2.secgroup.id}"
}
```

Please note that this behavior may differ depending on the configuration of the FlexibleEngine cloud. The above illustrates the current default Neutron behavior. Some FlexibleEngine clouds might provide additional rules and some might not provide any rules at all (in which case the `delete_default_rules` setting is moot).

Import

Security Groups can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_networking_secgroup_v2.secgroup_1 38809219-5e8a-4852-9139-6f461c90e8bc
```

flexibleengine_networking_subnet_v2

Manages a V2 Neutron subnet resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_networking_network_v2" "network_1" {
  name          = "tf_test_network"
  admin_state_up = "true"
}

resource "flexibleengine_networking_subnet_v2" "subnet_1" {
  network_id = "${flexibleengine_networking_network_v2.network_1.id}"
  cidr       = "192.168.199.0/24"
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to obtain the V2 Networking client. A Networking client is needed to create a Neutron subnet. If omitted, the `region` argument of the provider is used. Changing this creates a new subnet.
- `network_id` - (Required) The UUID of the parent network. Changing this creates a new subnet.
- `cidr` - (Required) CIDR representing IP range for this subnet, based on IP version. Changing this creates a new subnet.
- `ip_version` - (Optional) IP version, either 4 (default) or 6. Changing this creates a new subnet.
- `name` - (Optional) The name of the subnet. Changing this updates the name of the existing subnet.
- `tenant_id` - (Optional) The owner of the subnet. Required if admin wants to create a subnet for another tenant. Changing this creates a new subnet.
- `allocation_pools` - (Optional) An array of sub-ranges of CIDR available for dynamic allocation to ports. The `allocation_pool` object structure is documented below. Changing this creates a new subnet.
- `gateway_ip` - (Optional) Default gateway used by devices in this subnet. Leaving this blank and not setting `no_gateway` will cause a default gateway of `.1` to be used. Changing this updates the gateway IP of the existing subnet.
- `no_gateway` - (Optional) Do not set a gateway IP on this subnet. Changing this removes or adds a default gateway IP of the existing subnet.
- `enable_dhcp` - (Optional) The administrative state of the network. Acceptable values are "true" and "false". Changing this value enables or disables the DHCP capabilities of the existing subnet. Defaults to true.
- `dns_nameservers` - (Optional) An array of DNS name server names used by hosts in this subnet. Changing this updates the DNS name servers for the existing subnet.

- `host_routes` - (Optional) An array of routes that should be used by devices with IPs from this subnet (not including local subnet route). The `host_route` object structure is documented below. Changing this updates the host routes for the existing subnet.
- `value_specs` - (Optional) Map of additional options.

The `allocation_pools` block supports:

- `start` - (Required) The starting address.
- `end` - (Required) The ending addresss.

The `host_routes` block supports:

- `destination_cidr` - (Required) The destination CIDR.
- `next_hop` - (Required) The next hop in the route.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `network_id` - See Argument Reference above.
- `cidr` - See Argument Reference above.
- `ip_version` - See Argument Reference above.
- `name` - See Argument Reference above.
- `tenant_id` - See Argument Reference above.
- `allocation_pools` - See Argument Reference above.
- `gateway_ip` - See Argument Reference above.
- `enable_dhcp` - See Argument Reference above.
- `dns_nameservers` - See Argument Reference above.
- `host_routes` - See Argument Reference above.

Import

Subnets can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_networking_subnet_v2.subnet_1 da4faf16-5546-41e4-8330-4d0002b74048
```

flexibleengine_networking_vip_associate_v2

Manages a V2 vip associate resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_networking_network_v2" "network_1" {
  name = "network_1"
  admin_state_up = "true"
}

resource "flexibleengine_networking_subnet_v2" "subnet_1" {
  name = "subnet_1"
  cidr = "192.168.199.0/24"
  ip_version = 4
  network_id = "${flexibleengine_networking_network_v2.network_1.id}"
}

resource "flexibleengine_networking_router_interface_v2" "router_interface_1" {
  router_id = "${flexibleengine_networking_router_v2.router_1.id}"
  subnet_id = "${flexibleengine_networking_subnet_v2.subnet_1.id}"
}

resource "flexibleengine_networking_router_v2" "router_1" {
  name = "router_1"
  external_gateway = "0a2228f2-7f8a-45f1-8e09-9039e1d09975"
}

resource "flexibleengine_networking_port_v2" "port_1" {
  name = "port_1"
  admin_state_up = "true"
  network_id = "${flexibleengine_networking_network_v2.network_1.id}"

  fixed_ip {
    subnet_id = "${flexibleengine_networking_subnet_v2.subnet_1.id}"
  }
}

resource "flexibleengine_compute_instance_v2" "instance_1" {
  name = "instance_1"
  security_groups = ["default"]

  network {
    port = "${flexibleengine_networking_port_v2.port_1.id}"
  }
}

resource "flexibleengine_networking_port_v2" "port_2" {
  name = "port_2"
  admin_state_up = "true"
  network_id = "${flexibleengine_networking_network_v2.network_1.id}"

  fixed_ip {
    subnet_id = "${flexibleengine_networking_subnet_v2.subnet_1.id}"
  }
}
```

```
}

resource "flexibleengine_compute_instance_v2" "instance_2" {
  name = "instance_2"
  security_groups = ["default"]

  network {
    port = "${flexibleengine_networking_port_v2.port_1.id}"
  }
}

resource "flexibleengine_networking_vip_v2" "vip_1" {
  network_id = "${flexibleengine_networking_network_v2.network_1.id}"
  subnet_id = "${flexibleengine_networking_subnet_v2.subnet_1.id}"
}

resource "flexibleengine_networking_vip_associate_v2" "vip_associate_1" {
  vip_id = "${flexibleengine_networking_vip_v2.vip_1.id}"
  port_ids = ["${flexibleengine_networking_port_v2.port_1.id}", "${flexibleengine_networking_port_v2.port_2.id}"]
}
```

Argument Reference

The following arguments are supported:

- `vip_id` - (Required) The ID of vip to attach the port to. Changing this creates a new vip associate.
- `port_ids` - (Required) An array of one or more IDs of the ports to attach the vip to. Changing this creates a new vip associate.

Attributes Reference

The following attributes are exported:

- `vip_id` - See Argument Reference above.
- `port_ids` - See Argument Reference above.
- `vip_subnet_id` - The ID of the subnet this vip connects to.
- `vip_ip_address` - The IP address in the subnet for this vip.

flexibleengine_networking_vip_v2

Manages a V2 vip resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_networking_network_v2" "network_1" {
  name = "network_1"
  admin_state_up = "true"
}

resource "flexibleengine_networking_subnet_v2" "subnet_1" {
  name = "subnet_1"
  cidr = "192.168.199.0/24"
  ip_version = 4
  network_id = "${flexibleengine_networking_network_v2.network_1.id}"
}

resource "flexibleengine_networking_router_interface_v2" "router_interface_1" {
  router_id = "${flexibleengine_networking_router_v2.router_1.id}"
  subnet_id = "${flexibleengine_networking_subnet_v2.subnet_1.id}"
}

resource "flexibleengine_networking_router_v2" "router_1" {
  name = "router_1"
  external_gateway = "0a2228f2-7f8a-45f1-8e09-9039e1d09975"
}

resource "flexibleengine_networking_vip_v2" "vip_1" {
  network_id = "${flexibleengine_networking_network_v2.network_1.id}"
  subnet_id = "${flexibleengine_networking_subnet_v2.subnet_1.id}"
}
```

Argument Reference

The following arguments are supported:

- `network_id` - (Required) The ID of the network to attach the vip to. Changing this creates a new vip.
- `subnet_id` - (Required) Subnet in which to allocate IP address for this vip. Changing this creates a new vip.
- `ip_address` - (Optional) IP address desired in the subnet for this vip. If you don't specify `ip_address`, an available IP address from the specified subnet will be allocated to this vip.
- `name` - (Optional) A unique name for the vip.

Attributes Reference

The following attributes are exported:

- `network_id` - See Argument Reference above.
- `subnet_id` - See Argument Reference above.
- `ip_address` - See Argument Reference above.
- `name` - See Argument Reference above.
- `status` - The status of vip.
- `id` - The ID of the vip.
- `tenant_id` - The tenant ID of the vip.
- `device_owner` - The device owner of the vip.

flexibleengine_rds_instance_v1

Manages rds instance resource within FlexibleEngine

Example Usage: Creating a PostgreSQL RDS instance

```
data "flexibleengine_rds_flavors_v1" "flavor" {
  region = "eu-de"
  datastore_name = "PostgreSQL"
  datastore_version = "9.5.5"
  speccode = "rds.pg.s1.large.ha"
}

resource "flexibleengine_compute_secgroup_v2" "secgrp_rds" {
  name      = "secgrp-rds-instance"
  description = "Rds Security Group"
}

resource "flexibleengine_rds_instance_v1" "instance" {
  name = "rds-instance"
  datastore {
    type = "PostgreSQL"
    version = "9.5.5"
  }
  flavorref = "${data.flexibleengine_rds_flavors_v1.flavor.id}"
  volume {
    type = "COMMON"
    size = 200
  }
  region = "eu-de"
  availabilityzone = "eu-de-01"
  vpc = "c1095fe7-03df-4205-ad2d-6f4c181d436e"
  nics {
    subnetid = "b65f8d25-c533-47e2-8601-cfaa265a3e3e"
  }
  securitygroup {
    id = "${flexibleengine_compute_secgroup_v2.secgrp_rds.id}"
  }
  dbport = "8635"
  backupstrategy {
    starttime = "04:00:00"
    keepdays = 4
  }
  dbrtpd = "Huangwei!120521"
  ha {
    enable = true
    replicationmode = "async"
  }
  depends_on = ["flexibleengine_compute_secgroup_v2.secgrp_rds"]
}
```

Example Usage: Creating a SQLServer RDS instance

```
data "flexibleengine_rds_flavors_v1" "flavor" {
  region = "eu-west-0"
  datastore_name = "SQLServer"
  datastore_version = "2014 SP2 SE"
  speccode = "rds.mssql.s1.2xlarge"
}

resource "flexibleengine_compute_secgroup_v2" "secgrp_rds" {
  name      = "secgrp-rds-instance"
  description = "Rds Security Group"
}

resource "flexibleengine_rds_instance_v1" "instance" {
  name = "rds-instance"
  datastore {
    type = "SQLServer"
    version = "2014 SP2 SE"
  }
  flavorref = "${data.flexibleengine_rds_flavors_v1.flavor.id}"
  volume {
    type = "COMMON"
    size = 200
  }
  region = "eu-west-0"
  availabilityzone = "eu-west-0a"
  vpc = "c1095fe7-03df-4205-ad2d-6f4c181d436e"
  nics {
    subnetid = "b65f8d25-c533-47e2-8601-cfaa265a3e3e"
  }
  securitygroup {
    id = "${flexibleengine_compute_secgroup_v2.secgrp_rds.id}"
  }
  dbport = "8635"
  backupstrategy {
    starttime = "04:00:00"
    keepdays = 4
  }
  dbrtpd = "Huangwei!120521"
  depends_on = ["flexibleengine_compute_secgroup_v2.secgrp_rds"]
}
```

Example Usage: Creating a MySQL RDS instance

```

data "flexibleengine_rds_flavors_v1" "flavor" {
  region = "eu-west-0"
  datastore_name = "MySQL"
  datastore_version = "5.6.30"
  speccode = "rds.mysql.s1.medium"
}

resource "flexibleengine_compute_secgroup_v2" "secgrp_rds" {
  name      = "secgrp-rds-instance"
  description = "Rds Security Group"
}

resource "flexibleengine_rds_instance_v1" "instance" {
  name = "rds-instance"
  datastore {
    type = "MySQL"
    version = "5.6.30"
  }
  flavorref = "${data.flexibleengine_rds_flavors_v1.flavor.id}"
  volume {
    type = "COMMON"
    size = 200
  }
  region = "eu-west-0"
  availabilityzone = "eu-west-0a"
  vpc = "c1095fe7-03df-4205-ad2d-6f4c181d436e"
  nics {
    subnetid = "b65f8d25-c533-47e2-8601-cfaa265a3e3e"
  }
  securitygroup {
    id = "${flexibleengine_compute_secgroup_v2.secgrp_rds.id}"
  }
  dbport = "8635"
  backupstrategy {
    starttime = "04:00:00"
    keepdays = 4
  }
  dbbrtpd = "Huangwei!120521"
  ha {
    enable = true
    replicationmode = "async"
  }
  depends_on = ["flexibleengine_compute_secgroup_v2.secgrp_rds"]
}

```

Argument Reference

The following arguments are supported:

- **name** - (Required) Specifies the DB instance name. The DB instance name of the same type is unique in the same tenant. The changes of the instance name will be suppressed in HA scenario.
- **datastore** - (Required) Specifies database information. The structure is described below.

- `flavorref` - (Required) Specifies the specification ID (`flavors.id` in the response message in Obtaining All DB Instance Specifications). If you want to enable `ha` for the `rds` instance, a flavor with `ha speccode` is required.
- `volume` - (Required) Specifies the volume information. The structure is described below.
- `region` - (Required) Specifies the region ID.
- `availabilityzone` - (Required) Specifies the ID of the AZ.
- `vpc` - (Required) Specifies the VPC ID. For details about how to obtain this parameter value, see section "Virtual Private Cloud" in the Virtual Private Cloud API Reference.
- `nics` - (Required) Specifies the nics information. For details about how to obtain this parameter value, see section "Subnet" in the Virtual Private Cloud API Reference. The structure is described below.
- `securitygroup` - (Required) Specifies the security group which the RDS DB instance belongs to. The structure is described below.
- `dbport` - (Optional) Specifies the database port number.
- `backupstrategy` - (Optional) Specifies the advanced backup policy. The structure is described below.
- `dbrtpd` - (Required) Specifies the password for user `root` of the database.
- `ha` - (Optional) Specifies the parameters configured on HA and is used when creating HA DB instances. The structure is described below. NOTICE: RDS for Microsoft SQL Server does not support creating HA DB instances and this parameter is not involved.

The `datastore` block supports:

- `type` - (Required) Specifies the DB engine. Currently, MySQL, PostgreSQL, and Microsoft SQL Server are supported. The value is MySQL, PostgreSQL, or SQLServer.
- `version` - (Required) Specifies the DB instance version.
- Available value for attributes

<code>type</code>	<code>version</code>
PostgreSQL	9.5.5
	9.6.3
	9.6.5
MySQL	5.6.33
	5.6.30
	5.6.34
	5.6.35
	5.7.17
SQLServer	2014 SP2 SE

The `volume` block supports:

- `type` - (Required) Specifies the volume type. Valid value: It must be COMMON (SATA) or ULTRAHIGH (SSD) and is case-sensitive.
- `size` - (Required) Specifies the volume size. Its value must be a multiple of 10 and the value range is 100 GB to 2000 GB.

The `nics` block supports:

- `subnetId` - (Required) Specifies the subnet ID obtained from the VPC.

The `securitygroup` block supports:

- `id` - (Required) Specifies the ID obtained from the securitygroup.

The `backupstrategy` block supports:

- `starttime` - (Optional) Indicates the backup start time that has been set. The backup task will be triggered within one hour after the backup start time. Valid value: The value cannot be empty. It must use the hh:mm:ss format and must be valid. The current time is the UTC time.
- `keepdays` - (Optional) Specifies the number of days to retain the generated backup files. Its value range is 0 to 35. If this parameter is not specified or set to 0, the automated backup policy is disabled.

The `ha` block supports:

- `enable` - (Optional) Specifies the configured parameters on the HA. Valid value: The value is true or false. The value true indicates creating HA DB instances. The value false indicates creating a single DB instance.
- `replicationmode` - (Optional) Specifies the replication mode for the standby DB instance. The value cannot be empty. For MySQL, the value is `async` or `semisync`. For PostgreSQL, the value is `async` or `sync`.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `name` - See Argument Reference above.
- `flavorref` - See Argument Reference above.
- `volume` - See Argument Reference above.
- `availabilityzone` - See Argument Reference above.
- `vpc` - See Argument Reference above.
- `nics` - See Argument Reference above.
- `securitygroup` - See Argument Reference above.
- `dbport` - See Argument Reference above.
- `backupstrategy` - See Argument Reference above.
- `dbrtpd` - See Argument Reference above.
- `ha` - See Argument Reference above.
- `status` - Indicates the DB instance status.
- `hostname` - Indicates the instance connection address. It is a blank string.
- `type` - Indicates the DB instance type, which can be master or readreplica.

- `created` - Indicates the creation time in the following format: yyyy-mm-dd Thh:mm:ssZ.
- `updated` - Indicates the update time in the following format: yyyy-mm-dd Thh:mm:ssZ.

Attributes Reference

The following attributes can be updated:

- `volume.size` - See Argument Reference above.
- `flavorref` - See Argument Reference above.
- `backupstrategy` - See Argument Reference above.

flexibleengine_rds_instance_v3

instance management

Example Usage

create a single db instance

```
resource "flexibleengine_networking_secgroup_v2" "secgroup" {
  name = "terraform_test_security_group"
  description = "terraform security group acceptance test"
}

resource "flexibleengine_rds_instance_v3" "instance" {
  availability_zone = ["{{ availability_zone }}"]
  db {
    password = "Huangwei!120521"
    type = "PostgreSQL"
    version = "9.5"
    port = "8635"
  }
  name = "terraform_test_rds_instance"
  security_group_id = "${flexibleengine_networking_secgroup_v2.secgroup.id}"
  subnet_id = "{{ subnet_id }}"
  vpc_id = "{{ vpc_id }}"
  volume {
    type = "COMMON"
    size = 100
  }
  flavor = "rds.pg.s1.medium"
  backup_strategy {
    start_time = "08:00-09:00"
    keep_days = 1
  }
}
```

create a primary/standby db instance

```
resource "flexibleengine_networking_secgroup_v2" "secgroup" {
  name = "terraform_test_security_group"
  description = "terraform security group acceptance test"
}

resource "flexibleengine_rds_instance_v3" "instance" {
  availability_zone = ["{{ availability_zone_1 }}", "{{ availability_zone_2 }}"]
  db {
    password = "Huangwei!120521"
    type = "PostgreSQL"
    version = "9.5"
    port = "8635"
  }
  name = "terraform_test_rds_instance"
  security_group_id = "${flexibleengine_networking_secgroup_v2.secgroup.id}"
  subnet_id = "{{ subnet_id }}"
  vpc_id = "{{ vpc_id }}"
  volume {
    type = "COMMON"
    size = 100
  }
  flavor = "rds.pg.s1.medium.ha"
  ha_replication_mode = "async"
  backup_strategy {
    start_time = "08:00-09:00"
    keep_days = 1
  }
}
```

create a single db instance with encrypted volume

```

resource "flexibleengine_kms_key_v1" "key" {
  key_alias      = "key_1"
  key_description = "first test key"
  is_enabled     = true
}

resource "flexibleengine_networking_secgroup_v2" "secgroup" {
  name = "terraform_test_security_group"
  description = "terraform security group acceptance test"
}

resource "flexibleengine_rds_instance_v3" "instance" {
  availability_zone = ["{{ availability_zone }}"]
  db {
    password = "Huangwei!120521"
    type     = "PostgreSQL"
    version  = "9.5"
    port     = "8635"
  }
  name = "terraform_test_rds_instance"
  security_group_id = "${flexibleengine_networking_secgroup_v2.secgroup.id}"
  subnet_id = "{{ subnet_id }}"
  vpc_id = "{{ vpc_id }}"
  volume {
    disk_encryption_id = "${flexibleengine_kms_key_v1.key.id}"
    type = "COMMON"
    size = 100
  }
  flavor = "rds.pg.s1.medium"
  backup_strategy {
    start_time = "08:00-09:00"
    keep_days = 1
  }
}

```

Argument Reference

The following arguments are supported:

- `availability_zone` - (Required) Specifies the AZ name. Changing this parameter will create a new resource.
- `db` - (Required) Specifies the database information. Structure is documented below. Changing this parameter will create a new resource.
- `flavor` - (Required) Specifies the specification code. Changing this parameter will create a new resource.
- `name` - (Required) Specifies the DB instance name. The DB instance name of the same type must be unique for the same tenant. The value must be 4 to 64 characters in length and start with a letter. It is case-sensitive and can contain only letters, digits, hyphens (-), and underscores (_). Changing this parameter will create a new resource.
- `security_group_id` - (Required) Specifies the security group which the RDS DB instance belongs to. Changing this parameter will create a new resource.
- `subnet_id` - (Required) Specifies the subnet id. Changing this parameter will create a new resource.

- `volume` - (Required) Specifies the volume information. Structure is documented below. Changing this parameter will create a new resource.
- `vpc_id` - (Required) Specifies the VPC ID. Changing this parameter will create a new resource.
- `backup_strategy` - (Optional) Specifies the advanced backup policy. Structure is documented below. Changing this parameter will create a new resource.
- `ha_replication_mode` - (Optional) Specifies the replication mode for the standby DB instance. For MySQL, the value is `async` or `semisync`. For PostgreSQL, the value is `async` or `sync`. For Microsoft SQL Server, the value is `sync`. NOTE: `async` indicates the asynchronous replication mode. `semisync` indicates the semi-synchronous replication mode. `sync` indicates the synchronous replication mode. Changing this parameter will create a new resource.
- `param_group_id` - (Optional) Specifies the parameter group ID. Changing this parameter will create a new resource.

The `db` block supports:

- `password` - (Required) Specifies the database password. The value cannot be empty and should contain 8 to 32 characters, including uppercase and lowercase letters, digits, and the following special characters: `~!@#%*-+=~?` You are advised to enter a strong password to improve security, preventing security risks such as brute force cracking. Changing this parameter will create a new resource.
- `port` - (Optional) Specifies the database port information. The MySQL database port ranges from 1024 to 65535 (excluding 12017 and 33071, which are occupied by the RDS system and cannot be used). The PostgreSQL database port ranges from 2100 to 9500. The Microsoft SQL Server database port can be 1433 or ranges from 2100 to 9500, excluding 5355 and 5985. If this parameter is not set, the default value is as follows: For MySQL, the default value is 3306. For PostgreSQL, the default value is 5432. For Microsoft SQL Server, the default value is 1433. Changing this parameter will create a new resource.
- `type` - (Required) Specifies the DB engine. Value: MySQL, PostgreSQL, SQLServer. Changing this parameter will create a new resource.
- `user_name` - Indicates the default user name of database.
- `version` - (Required) Specifies the database version. MySQL databases support MySQL 5.6 and 5.7. PostgreSQL databases support PostgreSQL 9.5 and 9.6. Microsoft SQL Server databases support 2014 SE, 2016 SE, and 2016 EE. Changing this parameter will create a new resource.

The `volume` block supports:

- `disk_encryption_id` - (Optional) Specifies the key ID for disk encryption. Changing this parameter will create a new resource.
- `size` - (Required) Specifies the volume size. Its value range is from 40 GB to 4000 GB. The value must be a multiple of 10. Changing this parameter will create a new resource.
- `type` - (Required) Specifies the volume type. Its value can be any of the following and is case-sensitive: `COMMON`: indicates the SATA type. `ULTRAHIGH`: indicates the SSD type. Changing this parameter will create a new resource.

The `backup_strategy` block supports:

- `keep_days` - (Optional) Specifies the retention days for specific backup files. The value range is from 0 to 732. If this parameter is not specified or set to 0, the automated backup policy is disabled. NOTICE: Primary/standby DB instances of Microsoft SQL Server do not support disabling the automated backup policy. Changing this parameter will create a

new resource.

- `start_time` - (Required) Specifies the backup time window. Automated backups will be triggered during the backup time window. It must be a valid value in the "hh:mm-HH:MM" format. The current time is in the UTC format. The HH value must be 1 greater than the hh value. The values of mm and MM must be the same and must be set to any of the following: 00, 15, 30, or

1. Example value: 08:15-09:15 23:00-00:00. Changing this parameter will create a new resource.

Attributes Reference

In addition to the arguments listed above, the following computed attributes are exported:

- `created` - Indicates the creation time.
- `nodes` - Indicates the instance nodes information. Structure is documented below.
- `private_ips` - Indicates the private IP address list. It is a blank string until an ECS is created.
- `public_ips` - Indicates the public IP address list.

The `nodes` block contains:

- `availability_zone` - Indicates the AZ.
- `id` - Indicates the node ID.
- `name` - Indicates the node name.
- `role` - Indicates the node type. The value can be master or slave, indicating the primary node or standby node respectively.
- `status` - Indicates the node status.

Timeouts

This resource provides the following timeouts configuration options: - `create` - Default is 30 minute.

Import

RDS instance can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_rds_instance_v3.instance_1 7117d38e-4c8f-4624-a505-bd96b97d024c
```

But due to some attributes missing from the API response, it's required to ignore changes as below.

```
resource "flexibleengine_rds_instance_v3" "instance_1" {  
  ...  
  
  lifecycle {  
    ignore_changes = [  
      "db",  
    ]  
  }  
}
```

flexibleengine_rds_parametergroup_v3

Manages a V3 RDS parametergroup resource.

Example Usage

```
resource "flexibleengine_rds_parametergroup_v3" "pg_1" {
  name = "pg_1"
  description = "description_1"
  values = {
    max_connections = "10"
    autocommit = "OFF"
  }
  datastore {
    type = "mysql"
    version = "5.6"
  }
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The parameter group name. It contains a maximum of 64 characters.
- `description` - (Optional) The parameter group description. It contains a maximum of 256 characters and cannot contain the following special characters: >!<"&'= the value is left blank by default.
- `values` - (Optional) Parameter group values key/value pairs defined by users based on the default parameter groups.
- `datastore` - (Required) Database object. The database object structure is documented below. Changing this creates a new parameter group.

The `datastore` block supports:

- `type` - (Required) The DB engine. Currently, MySQL, PostgreSQL, and Microsoft SQL Server are supported. The value is case-insensitive and can be `mysql`, `postgresql`, or `sqlserver`.
- `version` - (Required) Specifies the database version.
 - MySQL databases support MySQL 5.6 and 5.7. Example value: 5.7.
 - PostgreSQL databases support PostgreSQL 9.5 and 9.6. Example value: 9.5.
 - Microsoft SQL Server databases support 2014 SE, 2016 SE, and 2016 EE. Example value: 2014_SE.

Attributes Reference

The following attributes are exported:

- `id` - ID of the parameter group.
- `configuration_parameters` - Indicates the parameter configuration defined by users based on the default parameters groups.
 - `name` - Indicates the parameter name.
 - `value` - Indicates the parameter value.
 - `restart_required` - Indicates whether a restart is required.
 - `readonly` - Indicates whether the parameter is read-only.
 - `value_range` - Indicates the parameter value range.
 - `type` - Indicates the parameter type.
 - `description` - Indicates the parameter description.

Import

Parameter groups can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_rds_parametergroup_v3.pg_1 7117d38e-4c8f-4624-a505-bd96b97d024c
```

flexibleengine_rts_software_config_v1

Provides an RTS software config resource.

Example Usage

```
variable "config_name" {}

resource "flexibleengine_rts_software_config_v1" "myconfig" {
  name = "${var.config_name}"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the software configuration.
- `group` - (Optional) The namespace that groups this software configuration by when it is delivered to a server.
- `input_values` - (Optional) A list of software configuration inputs.
- `output_values` - (Optional) A list of software configuration outputs.
- `config` - (Optional) The software configuration code.
- `options` - (Optional) The software configuration options.

Attributes Reference

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

- `id` - The id of the software config.

Import

Software Config can be imported using the `config id`, e.g. `$ terraform import flexibleengine_rts_software_config_v1 4779ab1c-7c1a-44b1-a02e-93dfc361b32d`

flexibleengine_rts_stack_v1

Provides an FlexibleEngine Stack.

Example Usage

```
variable "name" { }
variable "network_id" { }
variable "instance_type" { }
variable "image_id" { }

resource "flexibleengine_rts_stack_v1" "mystack" {
  name = "${var.name}"
  disable_rollback = true
  timeout_mins=60
  parameters = {
    "network_id" = "${var.network_id}"
    "instance_type" = "${var.instance_type}"
    "image_id" = "${var.image_id}"
  }
  template_body = <<STACK
  {
    "heat_template_version": "2016-04-08",
    "description": "Simple template to deploy",
    "parameters": {
      "image_id": {
        "type": "string",
        "description": "Image to be used for compute instance",
        "label": "Image ID"
      },
      "network_id": {
        "type": "string",
        "description": "The Network to be used",
        "label": "Network UUID"
      },
      "instance_type": {
        "type": "string",
        "description": "Type of instance (Flavor) to be used",
        "label": "Instance Type"
      }
    }
  },
  "resources": {
    "my_instance": {
      "type": "OS::Nova::Server",
      "properties": {
        "image": {
          "get_param": "image_id"
        },
        "flavor": {
          "get_param": "instance_type"
        },
        "networks": [{
          "network": {
            "get_param": "network_id"
          }
        }
      ]
    }
  }
}
```

```

        ]]
      }
    }
  },
  "outputs": {
    "InstanceIP":{
      "description": "Instance IP",
      "value": { "get_attr": ["my_instance", "first_address"] }
    }
  }
}
STACK
}

```

Argument Reference

The following arguments are supported:

- `name` - (Required) A unique name for the stack. The value must meet the regular expression rule (`^[a-zA-Z][a-zA-Z0-9_.-]{0,254}$`). Changing this creates a new stack.
- `template_body` - (Optional; Required if `template_url` is empty) Structure containing the template body. The template content must use the yaml syntax.
- `template_url` - (Optional; Required if `template_body` is empty) Location of a file containing the template body.
- `environment` - (Optional) Tthe environment information about the stack.
- `files` - (Optional) Files used in the environment.
- `parameters` - (Optional) A list of Parameter structures that specify input parameters for the stack.
- `disable_rollback` - (Optional) Set to true to disable rollback of the stack if stack creation failed.
- `timeout_mins` - (Optional) Specifies the timeout duration.

Attributes Reference

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

- `outputs` - A map of outputs from the stack.
- `capabilities` - List of stack capabilities for stack.
- `notification_topics` - List of notification topics for stack.
- `status` - Specifies the stack status.

Import

RTS Stacks can be imported using the `name` , e.g.

```
$ terraform import flexibleengine_rts_stack_v1.mystack rts-stack
```

Timeouts

flexibleengine_rts_stack_v1 provides the following Timeouts (</docs/configuration/resources.html#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.

flexibleengine_s3_bucket

Provides a S3 bucket resource.

Example Usage

Private Bucket w/ Tags

```
resource "flexibleengine_s3_bucket" "b" {
  bucket = "my-tf-test-bucket"
  acl    = "private"
}
```

Static Website Hosting

```
resource "flexibleengine_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 "flexibleengine_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 "flexibleengine_s3_bucket" "b" {
  bucket = "my-tf-test-bucket"
  acl    = "private"

  versioning {
    enabled = true
  }
}
```

Enable Logging

```
resource "flexibleengine_s3_bucket" "log_bucket" {
  bucket = "my-tf-log-bucket"
  acl    = "log-delivery-write"
}

resource "flexibleengine_s3_bucket" "b" {
  bucket = "my-tf-test-bucket"
  acl    = "private"

  logging {
    target_bucket = "${flexibleengine_s3_bucket.log_bucket.id}"
    target_prefix = "log/"
  }
}
```

Using object lifecycle

```

resource "flexibleengine_s3_bucket" "bucket" {
  bucket = "my-bucket"
  acl    = "private"

  lifecycle_rule {
    id      = "log"
    enabled = true

    prefix = "log/"

    expiration {
      days = 90
    }
  }

  lifecycle_rule {
    id      = "tmp"
    prefix = "tmp/"
    enabled = true

    expiration {
      date = "2016-01-12"
    }
  }
}

resource "flexibleengine_s3_bucket" "versioning_bucket" {
  bucket = "my-versioning-bucket"
  acl    = "private"

  versioning {
    enabled = true
  }

  lifecycle_rule {
    prefix = "config/"
    enabled = true
  }
}

```

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 (<https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#canned-acl>) to apply. Defaults to "private".
- `policy` - (Optional) A valid bucket policy (<https://docs.aws.amazon.com/AmazonS3/latest/dev/example-bucket->

policies.html) 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.

- `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 (<https://docs.aws.amazon.com/AmazonS3/latest/dev/cors.html>) (documented below).
- `versioning` - (Optional) A state of versioning (<https://docs.aws.amazon.com/AmazonS3/latest/dev/Versioning.html>) (documented below)
- `logging` - (Optional) A settings of bucket logging (<https://docs.aws.amazon.com/AmazonS3/latest/UG/ManagingBucketLogging.html>) (documented below).
- `lifecycle_rule` - (Optional) A configuration of object lifecycle management (<http://docs.aws.amazon.com/AmazonS3/latest/dev/object-lifecycle-mgmt.html>) (documented below).
- `region` - (Optional) If specified, the region this bucket should reside in. Otherwise, the region used by the callee.

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 (<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/aws-properties-s3-websiteconfiguration-routingrules.html>) describing redirect behavior and when redirects are applied.

The `cors_rule` 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.
- `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).
- `noncurrent_version_expiration` - (Optional) Specifies when noncurrent object versions expire (documented below).

At least one of `expiration`, `noncurrent_version_expiration` 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 `noncurrent_version_expiration` object supports the following

- `days` (Required) Specifies the number of days an object is noncurrent object versions expire.

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
(https://docs.aws.amazon.com/general/latest/gr/rande.html#s3_website_region_endpoints) for this bucket's region.
- `region` - The 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 flexibleengine_s3_bucket.bucket bucket-name
```

flexibleengine_s3_bucket_object

Provides a S3 bucket object resource.

Example Usage

Uploading a file to a bucket

```
resource "flexibleengine_s3_bucket_object" "object" {
  bucket = "your_bucket_name"
  key    = "new_object_key"
  source = "path/to/file"
  etag   = "${md5(file("path/to/file"))}"
}

resource "flexibleengine_s3_bucket" "examplebucket" {
  bucket = "examplebuckettftest"
  acl    = "private"
}

resource "flexibleengine_s3_bucket_object" "examplebucket_object" {
  key          = "someobject"
  bucket       = "${flexibleengine_s3_bucket.examplebucket.bucket}"
  source       = "index.html"
}
```

Server Side Encryption with S3 Default Master Key

```
resource "flexibleengine_s3_bucket" "examplebucket" {
  bucket = "examplebuckettftest"
  acl    = "private"
}

resource "flexibleengine_s3_bucket_object" "examplebucket_object" {
  key          = "someobject"
  bucket       = "${flexibleengine_s3_bucket.examplebucket.bucket}"
  source       = "index.html"
  server_side_encryption = "aws:kms"
}
```

Argument Reference

Note: If you specify `content_encoding` you are responsible for encoding the body appropriately (i.e. `source` and `content` both 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) The path to the source file being uploaded to the bucket.
- `content` - (Required unless `source` given) The literal content being uploaded to the bucket.
- `acl` - (Optional) The canned ACL (<https://docs.aws.amazon.com/AmazonS3/latest/dev/acl-overview.html#canned-acl>) to apply. Defaults to "private".
- `cache_control` - (Optional) Specifies caching behavior along the request/reply chain. Read [w3c cache_control](http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.9) (<http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.9>) for further details.
- `content_disposition` - (Optional) Specifies presentational information for the object. Read [w3c content_disposition](http://www.w3.org/Protocols/rfc2616/rfc2616-sec19.html#sec19.5.1) (<http://www.w3.org/Protocols/rfc2616/rfc2616-sec19.html#sec19.5.1>) 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](http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.11) (<http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.11>) 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 (<http://docs.aws.amazon.com/AmazonS3/latest/dev/how-to-page-redirect.html>).
- `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".

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.

flexibleengine_s3_bucket_policy

Attaches a policy to an S3 bucket resource.

Example Usage

Basic Usage

```
resource "flexibleengine_s3_bucket" "b" {
  bucket = "my_tf_test_bucket"
}

resource "flexibleengine_s3_bucket_policy" "b" {
  bucket = "${flexibleengine_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.

flexibleengine_sfs_file_system_v2

Provides an Shared File System (SFS) resource.

Example Usage

```
variable "share_name" { }

variable "share_description" { }

variable "vpc_id" { }

resource "flexibleengine_sfs_file_system_v2" "share-file"
{
    size = 50
    name = "${var.share_name}"
    access_to = "${var.vpc_id}"
    access_level = "rw"
    description = "${var.share_description}"
    metadata = {
        "type"="nfs"
    }
}
```

Argument Reference

The following arguments are supported:

- `size` - (Required) The size (GB) of the shared file system.
- `share_proto` - (Optional) The protocol for sharing file systems. The default value is NFS.
- `name` - (Optional) The name of the shared file system.
- `description` - (Optional) Describes the shared file system.
- `is_public` - (Optional) The level of visibility for the shared file system.
- `metadata` - (Optional) Metadata key and value pairs as a dictionary of strings.Changing this will create a new resource.
- `availability_zone` - (Optional) The availability zone name.Changing this parameter will create a new resource.
- `access_level` - (Required) The access level of the shared file system. Changing this will create a new access rule.
- `access_type` - (Optional) The type of the share access rule. Changing this will create a new access rule.
- `access_to` - (Required) The access that the back end grants or denies. Changing this will create a new access rule

Attributes Reference

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

- `id` - The UUID of the shared file system.
- `status` - The status of the shared file system.
- `share_type` - The storage service type assigned for the shared file system, such as high-performance storage (composed of SSDs) and large-capacity storage (composed of SATA disks).
- `volume_type` - The volume type.
- `export_location` - The address for accessing the shared file system.
- `host` - The host name of the shared file system.
- `share_access_id` - The UUID of the share access rule.
- `access_rules_status` - The status of the share access rule.

Import

SFS can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_sfs_file_system_v2 4779ab1c-7c1a-44b1-a02e-93dfc361b32d
```

flexibleengine_smn_subscription_v2

Manages a V2 subscription resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_smn_topic_v2" "topic_1" {
  name          = "topic_1"
  display_name  = "The display name of topic_1"
}

resource "flexibleengine_smn_subscription_v2" "subscription_1" {
  topic_urn      = "${flexibleengine_smn_topic_v2.topic_1.id}"
  endpoint       = "mailto:mailtest@gmail.com"
  protocol       = "email"
  remark        = "O&M"
}

resource "flexibleengine_smn_subscription_v2" "subscription_2" {
  topic_urn      = "${flexibleengine_smn_topic_v2.topic_1.id}"
  endpoint       = "13600000000"
  protocol       = "sms"
  remark        = "O&M"
}
```

Argument Reference

The following arguments are supported:

- `topic_urn` - (Required) Resource identifier of a topic, which is unique.
- `endpoint` - (Required) Message endpoint. For an HTTP subscription, the endpoint starts with `http://`. For an HTTPS subscription, the endpoint starts with `https://`. For an email subscription, the endpoint is a mail address. For an SMS message subscription, the endpoint is a phone number.
- `protocol` - (Required) Protocol of the message endpoint. Currently, `email`, `sms`, `http`, and `https` are supported.
- `remark` - (Optional) Remark information. The remarks must be a UTF-8-coded character string containing 128 bytes.
- `subscription_urn` - (Optional) Resource identifier of a subscription, which is unique.
- `owner` - (Optional) Project ID of the topic creator.
- `status` - (Optional) Subscription status. 0 indicates that the subscription is not confirmed. 1 indicates that the subscription is confirmed. 3 indicates that the subscription is canceled.

Attributes Reference

The following attributes are exported:

- `topic_urn` - See Argument Reference above.
- `endpoint` - See Argument Reference above.
- `protocol` - See Argument Reference above.
- `remark` - See Argument Reference above.
- `subscription_urn` - See Argument Reference above.
- `owner` - See Argument Reference above.
- `status` - See Argument Reference above.

flexibleengine_smn_topic_v2

Manages a V2 topic resource within FlexibleEngine.

Example Usage

```
resource "flexibleengine_smn_topic_v2" "topic_1" {  
  name          = "topic_1"  
  display_name  = "The display name of topic_1"  
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the topic to be created.
- `display_name` - (Optional) Topic display name, which is presented as the name of the email sender in an email message.

Attributes Reference

The following attributes are exported:

- `name` - See Argument Reference above.
- `display_name` - See Argument Reference above.
- `topic_urn` - Resource identifier of a topic, which is unique.
- `push_policy` - Message pushing policy. 0 indicates that the message sending fails and the message is cached in the queue. 1 indicates that the failed message is discarded.
- `create_time` - Time when the topic was created.
- `update_time` - Time when the topic was updated.

flexibleengine_vbs_backup_policy_v2

Provides an VBS Backup Policy resource.

Example Usage

```
resource "flexibleengine_vbs_backup_policy_v2" "vbs" {  
  name = "policy_002"  
  start_time = "12:00"  
  status = "ON"  
  retain_first_backup = "N"  
  retention_num = 2  
  frequency = 1  
}
```

Argument Reference

The following arguments are supported:

- `name` (Required) - Specifies the policy name. The value is a string of 1 to 64 characters that can contain letters, digits, underscores (`_`), and hyphens (`-`). It cannot start with default.
- `start_time` (Required) - Specifies the start time of the backup job. The value is in the HH:mm format.
- `status` (Required) - Specifies the backup policy status. The value can ON or OFF.
- `retain_first_backup` (Required) - Specifies whether to retain the first backup in the current month. Possible values are Y or N.
- `retention_num` (Required) - Specifies number of retained backups. Minimum value is 2.
- `frequency` (Required) - Specifies the backup interval. The value is in the range of 1 to 14 days.

Attributes Reference

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

- `id` - Specifies a backup policy ID.
- `policy_resource_count` - Specifies the number of volumes associated with the backup policy.

Import

Backup Policy can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_vbs_backup_policy_v2.vbs 4779ab1c-7c1a-44b1-a02e-93dfc361b32d
```

flexibleengine_vbs_backup_v2

Provides an VBS Backup resource.

Example Usage

```
variable "backup_name" {}

variable "volume_id" {}

resource "flexibleengine_vbs_backup_v2" "mybackup" {
  volume_id = "${var.volume_id}"
  name      = "${var.backup_name}"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the vbs backup. Changing the parameter will create new resource.
- `volume_id` - (Required) The id of the disk to be backed up. Changing the parameter will create new resource.
- `snapshot_id` - (Optional) The snapshot id of the disk to be backed up. Changing the parameter will create new resource.
- `description` - (Optional) The description of the vbs backup. Changing the parameter will create new resource.

Attributes Reference

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

- `id` - The id of the vbs backup.
- `container` - The container of the backup.
- `status` - The status of the VBS backup.
- `availability_zone` - The AZ where the backup resides.
- `size` - The size of the vbs backup.
- `service_metadata` - The metadata of the vbs backup.

Import

VBS Backup can be imported using the `backup id`, e.g.

```
$ terraform import flexibleengine_vbs_backup_v2.mybackup 4779ab1c-7c1a-44b1-a02e-93dfc361b32d
```

flexibleengine_vpc_eip_v1

Manages a V1 EIP resource within FlexibleEngine VPC.

Example Usage

```
resource "flexibleengine_vpc_eip_v1" "eip_1" {
  publicip {
    type = "5_bgp"
  }
  bandwidth {
    name = "test"
    size = 8
    share_type = "PER"
    charge_mode = "traffic"
  }
}
```

Argument Reference

The following arguments are supported:

- `region` - (Optional) The region in which to create the eip. If omitted, the `region` argument of the provider is used. Changing this creates a new eip.
- `publicip` - (Required) The elastic IP address object.
- `bandwidth` - (Required) The bandwidth object.

The `publicip` block supports:

- `type` - (Required) The value must be a type supported by the system. Only `5_bgp` supported now. Changing this creates a new eip.
- `ip_address` - (Optional) The value must be a valid IP address in the available IP address segment. Changing this creates a new eip.
- `port_id` - (Optional) The port id which this eip will associate with. If the value is "" or this not specified, the eip will be in unbind state.

The `bandwidth` block supports:

- `name` - (Required) The bandwidth name, which is a string of 1 to 64 characters that contain letters, digits, underscores (`_`), and hyphens (`-`).
- `size` - (Required) The bandwidth size. The value ranges from 1 to 300 Mbit/s.
- `share_type` - (Required) Whether the bandwidth is shared or exclusive. Changing this creates a new eip.
- `charge_mode` - (Optional) This is a reserved field. If the system supports charging by traffic and this field is specified, then you are charged by traffic for elastic IP addresses. Changing this creates a new eip.

Attributes Reference

The following attributes are exported:

- `region` - See Argument Reference above.
- `publicip/type` - See Argument Reference above.
- `publicip/ip_address` - See Argument Reference above.
- `publicip/port_id` - See Argument Reference above.
- `bandwidth/name` - See Argument Reference above.
- `bandwidth/size` - See Argument Reference above.
- `bandwidth/share_type` - See Argument Reference above.
- `bandwidth/charge_mode` - See Argument Reference above.

Import

EIPs can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_vpc_eip_v1.eip_1 2c7f39f3-702b-48d1-940c-b50384177ee1
```

flexibleengine_vpc_peering_connection_accepter_v2

Provides a resource to manage the acceptor's side of a VPC Peering Connection.

When a cross-tenant (requester's tenant differs from the acceptor's tenant) VPC Peering Connection is created, a VPC Peering Connection resource is automatically created in the acceptor's account. The requester can use the `flexibleengine_vpc_peering_connection_v2` resource to manage its side of the connection and the acceptor can use the `flexibleengine_vpc_peering_connection_accepter_v2` resource to "adopt" its side of the connection into management.

Example Usage

```

provider "flexibleengine" {
  alias = "main"
  user_name = "${var.username}"
  domain_name = "${var.domain_name}"
  password = "${var.password}"
  auth_url = "${var.auth_url}"
  region = "${var.region}"
  tenant_id = "${var.tenant_id}"
}

provider "flexibleengine" {
  alias = "peer"
  user_name = "${var.peer_username}"
  domain_name = "${var.peer_domain_name}"
  password = "${var.peer_password}"
  auth_url = "${var.peer_auth_url}"
  region = "${var.peer_region}"
  tenant_id = "${var.peer_tenant_id}"
}

resource "flexibleengine_vpc_v1" "vpc_main" {
  provider = "flexibleengine.main"
  name = "${var.vpc_name}"
  cidr = "${var.vpc_cidr}"
}

resource "flexibleengine_vpc_v1" "vpc_peer" {
  provider = "flexibleengine.peer"
  name = "${var.peer_vpc_name}"
  cidr = "${var.peer_vpc_cidr}"
}

# Requester's side of the connection.
resource "flexibleengine_vpc_peering_connection_v2" "peering" {
  provider = "flexibleengine.main"
  name = "${var.peer_name}"
  vpc_id = "${flexibleengine_vpc_v1.vpc_main.id}"
  peer_vpc_id = "${flexibleengine_vpc_v1.vpc_peer.id}"
  peer_tenant_id = "${var.tenant_id}"
}

# Acceptor's side of the connection.
resource "flexibleengine_vpc_peering_connection_accepter_v2" "peer" {
  provider = "flexibleengine.peer"
  vpc_peering_connection_id = "${flexibleengine_vpc_peering_connection_v2.peering.id}"
  accept = true
}

```

Argument Reference

The following arguments are supported:

- `vpc_peering_connection_id` (Required) - The VPC Peering Connection ID to manage. Changing this creates a new VPC peering connection acceptor.

- `accept` (Optional)- Whether or not to accept the peering request. Defaults to `false`.

Removing `flexibleengine_vpc_peering_connection_accepter_v2` from your configuration

FlexibleEngine allows a cross-tenant 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 `flexibleengine_vpc_peering_connection_v2` resource from your configuration. Removing a `flexibleengine_vpc_peering_connection_accepter_v2` resource from your configuration will remove it from your state file and management, but will not destroy the VPC Peering Connection.

Attributes Reference

All of the argument attributes except `accept` are also exported as result attributes.

- `name` - The VPC peering connection name.
- `id` - The VPC peering connection ID.
- `status` - The VPC peering connection status.
- `vpc_id` - The ID of requester VPC involved in a VPC peering connection.
- `peer_vpc_id` - The VPC ID of the acceptor tenant.
- `peer_tenant_id` - The Tenant Id of the acceptor tenant.

flexibleengine_vpc_peering_connection_v2

Provides a resource to manage a VPC Peering Connection resource.

Note: For cross-tenant (requester's tenant differs from the acceptor's tenant) VPC Peering Connections, use the `flexibleengine_vpc_peering_connection_v2` resource to manage the requester's side of the connection and use the `flexibleengine_vpc_peering_connection_accepter_v2` resource to manage the acceptor's side of the connection.

Example Usage

```
resource "flexibleengine_vpc_peering_connection_v2" "peering" {
  name = "${var.peer_conn_name}"
  vpc_id = "${var.vpc_id}"
  peer_vpc_id = "${var.accepter_vpc_id}"
}
```

Argument Reference

The following arguments are supported:

- `name` (Required) - Specifies the name of the VPC peering connection. The value can contain 1 to 64 characters.
- `vpc_id` (Required) - Specifies the ID of a VPC involved in a VPC peering connection. Changing this creates a new VPC peering connection.
- `peer_vpc_id` (Required) - Specifies the VPC ID of the acceptor tenant. Changing this creates a new VPC peering connection.
- `peer_tenant_id` (Optional) - Specified the Tenant Id of the acceptor tenant. Changing this creates a new VPC peering connection.

Attributes Reference

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

- `id` - The VPC peering connection ID.
- `status` - The VPC peering connection status. The value can be `PENDING_ACCEPTANCE`, `REJECTED`, `EXPIRED`, `DELETED`, or `ACTIVE`.

Notes

If you create a VPC peering connection with another VPC of your own, the connection is created without the need for you to accept the connection.

Import

VPC Peering resources can be imported using the `vpc_peering_id`, e.g.

```
$ terraform import flexibleengine_vpc_peering_connection_v2.test_connection 22b76469-08e3-4937-8c1d-7aad34892be1
```

flexibleengine_vpc_route_v2

Provides a resource to create a route.

Example Usage

```
resource "flexibleengine_vpc_route_v2" "vpc_route" {  
  type = "peering"  
  nexthop = "${var.nexthop}"  
  destination = "192.168.0.0/16"  
  vpc_id = "${var.vpc_id}"  
}
```

Argument Reference

The following arguments are supported:

- `destination` (Required) - Specifies the destination IP address or CIDR block. Changing this creates a new Route.
- `nexthop` (Required) - Specifies the next hop. If the route type is peering, enter the VPC peering connection ID. Changing this creates a new Route.
- `type` (Required) - Specifies the route type. Currently, the value can only be **peering**. Changing this creates a new Route.
- `vpc_id` (Required) - Specifies the VPC for which a route is to be added. Changing this creates a new Route.
- `tenant_id` (Optional) - Specifies the tenant ID. Only the administrator can specify the tenant ID of other tenant. Changing this creates a new Route.

Attributes Reference

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

- `id` - The route ID.

flexibleengine_vpc_subnet_v1

Provides an VPC subnet resource.

Example Usage

```
resource "flexibleengine_vpc_v1" "vpc_v1" {
  name = "${var.vpc_name}"
  cidr = "${var.vpc_cidr}"
}

resource "flexibleengine_vpc_subnet_v1" "subnet_v1" {
  name = "${var.subnet_name}"
  cidr = "${var.subnet_cidr}"
  gateway_ip = "${var.subnet_gateway_ip}"
  vpc_id = "${flexibleengine_vpc_v1.vpc_v1.id}"
}
```

Argument Reference

The following arguments are supported:

- `name` (Required) - The subnet name. The value is a string of 1 to 64 characters that can contain letters, digits, underscores (`_`), and hyphens (`-`).
- `cidr` (Required) - Specifies the network segment on which the subnet resides. The value must be in CIDR format. The value must be within the CIDR block of the VPC. The subnet mask cannot be greater than 28. Changing this creates a new Subnet.
- `gateway_ip` (Required) - Specifies the gateway of the subnet. The value must be a valid IP address. The value must be an IP address in the subnet segment. Changing this creates a new Subnet.
- `vpc_id` (Required) - Specifies the ID of the VPC to which the subnet belongs. Changing this creates a new Subnet.
- `dhcp_enable` (Optional) - Specifies whether the DHCP function is enabled for the subnet. The value can be true or false. If this parameter is left blank, it is set to true by default.
- `primary_dns` (Optional) - Specifies the IP address of DNS server 1 on the subnet. The value must be a valid IP address.
- `secondary_dns` (Optional) - Specifies the IP address of DNS server 2 on the subnet. The value must be a valid IP address.
- `dns_list` (Optional) - Specifies the DNS server address list of a subnet. This field is required if you need to use more than two DNS servers. This parameter value is the superset of both DNS server address 1 and DNS server address 2.
- `availability_zone` (Optional) - Identifies the availability zone (AZ) to which the subnet belongs. The value must be an existing AZ in the system. Changing this creates a new Subnet.

Attributes Reference

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

- `id` - Specifies a resource ID in UUID format.
- `status` - Specifies the status of the subnet. The value can be ACTIVE, DOWN, UNKNOWN, or ERROR.
- `subnet_id` - Specifies the subnet (Native OpenStack API) ID.

Import

Subnets can be imported using the `subnet_id`, e.g.

```
$ terraform import flexibleengine_vpc_subnet_v1 4779ab1c-7c1a-44b1-a02e-93dfc361b32d
```

flexibleengine_vpc_v1

Manages a VPC resource within FlexibleEngine.

Example Usage

```
variable "vpc_name" {
  default = "flexibleengine_vpc"
}

variable "vpc_cidr" {
  default = "192.168.0.0/16"
}

resource "flexibleengine_vpc_v1" "vpc_v1" {
  name = "${var.vpc_name}"
  cidr = "${var.vpc_cidr}"
}
```

Argument Reference

The following arguments are supported:

- `cidr` - (Required) The range of available subnets in the VPC. The value ranges from 10.0.0.0/8 to 10.255.255.0/24, 172.16.0.0/12 to 172.31.255.0/24, or 192.168.0.0/16 to 192.168.255.0/24.
- `region` - (Optional) The region in which to obtain the V1 VPC client. A VPC client is needed to create a VPC. If omitted, the region argument of the provider is used. Changing this creates a new VPC.
- `name` - (Required) The name of the VPC. The name must be unique for a tenant. The value is a string of no more than 64 characters and can contain digits, letters, underscores (`_`), and hyphens (`-`). Changing this updates the name of the existing VPC.

Attributes Reference

The following attributes are exported:

- `id` - ID of the VPC.
- `name` - See Argument Reference above.
- `cidr` - See Argument Reference above.
- `status` - The current status of the desired VPC. Can be either `CREATING`, `OK`, `DOWN`, `PENDING_UPDATE`, `PENDING_DELETE`, or `ERROR`.

- `shared` - Specifies whether the cross-tenant sharing is supported.
- `region` - See Argument Reference above.

Import

VPCs can be imported using the `id`, e.g.

```
$ terraform import flexibleengine_vpc_v1.vpc_v1 7117d38e-4c8f-4624-a505-bd96b97d024c
```