» ucloud_projects

This data source providers a list of projects owned by user according to finance permission and name.

» Example Usage

```
data "ucloud_projects" "example" {
  is_finance = false
}

output "first" {
  value = data.ucloud_projects.example.projects[0].id
}
```

» Argument Reference

The following arguments are supported:

- `is_finance` - (Optional) To identify if the current account is granted with financial permission.
- `name_regex` - (Optional) A regex string to filter resulting projects by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `projects` - It is a nested type which documented below.
- `total_count` - Total number of projects that satisfy the condition.

The attribute (`projects`) support the following:

- `id` - The ID of project defined.
- `name` - The name of the defined project.
- `parent_id` - The ID of the parent project where the sub project belongs to.
- `parent_name` - The name of the parent project where the sub project belongs to.
- `member_count` - The number of members belongs to the defined project.

- `resource_count` - The number of the resounce instance belong/s to the defined project.
- `create_time` - The time of creation for instance, formatted in RFC3339 time string.

» ucloud_zones

This data source provides a list of available zones in the current region.

» Example Usage

```
data "ucloud_zones" "example" {}

output "first" {
  value = data.ucloud_zones.example.zones[0].id
}
```

» Argument Reference

The following arguments are supported:

- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `zones` - It is a nested type which documented below.
- `total_count` - Total number of zones that satisfy the condition.

The attribute (`zones`) support the following:

- `id` - The ID of availability zone.

» ucloud_images

This data source providers a list of available image resources according to their availability zone, image ID and other fields.

» Example Usage

```
data "ucloud_images" "example" {
  availability_zone = "cn-bj2-02"
  image_type        = "base"
  name_regex        = "^CentOS 7.[1-2] 64"
  most_recent       = true
}

output "first" {
  value = data.ucloud_images.example.images[0].id
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Optional) Availability zone where images are located. such as: `cn-bj2-02`. You may refer to list of availability zone.
- `name_regex` - (Optional) A regex string to filter resulting images by name. (Such as: `^CentOS 7.[1-2] 64` means CentOS 7.1 of 64-bit operating system or CentOS 7.2 of 64-bit operating system, "Ubuntu 16.04 64" means Ubuntu 16.04 of 64-bit operating system).
- `image_type` - (Optional) The type of image. Possible values are: `base` as standard image, `business` as owned by market place, and `custom` as custom-image, all the image types will be retrieved by default.
- `os_type` - (Optional) The type of OS. Possible values are: `linux` and `windows`, all the OS types will be retrieved by default.
- `most_recent` - (Optional) If more than one result is returned, use the most recent image.
- `image_id` - (Optional) The ID of image. ~> **Note** this argument conflicts with `ids`.
- `ids` - (Optional) A list of image IDs, all the images belong to this region will be retrieved if the ID is `[]`. ~> **Note** this argument conflicts with `image_id`.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `images` - It is a nested type which documented below.
- `total_count` - Total number of images that satisfy the condition.

The attribute (`images`) support the following:

- `availability_zone` - Availability zone where image is located.
- `create_time` - The time of creation for image, formatted in RFC3339 time string.
- `features` - To identify if any particular feature belongs to the instance, the value is `NetEnhnced` as I/O enhanced instance for now.
- `description` - The description of image if any.
- `id` - The ID of image.
- `name` - The name of image.
- `size` - The size of image.
- `type` - The type of image.
- `os_name` - The name of OS.
- `os_type` - The type of OS.
- `status` - The status of image. Possible values are `Available`, `Making` and `Unavailable`.

» ucloud_instances

This data source providers a list of UHost instance resources according to their availability zone, instance ID and tag.

» Example Usage

```
data "ucloud_instances" "example" {
  availability_zone = "cn-bj2-02"
}

output "first" {
  value = data.ucloud_instances.example.instances[0].id
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Optional) Availability zone where instances are located. Such as: "cn-bj2-02". You may refer to list of availability zone
- `ids` - (Optional) A list of instance IDs, all the instances belongs to the defined region will be retrieved if this argument is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting instances by name.

- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).
- `tag` - (Optional) A tag assigned to instance.

» Attributes Reference

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

- `instances` - It is a nested type. instances documented below.
- `total_count` - Total number of instances that satisfy the condition.

The attribute (`instances`) support the following:

- `availability_zone` - Availability zone where instances are located.
- `id` - The ID of instance.
- `name` - The name of the instance.
- `cpu` - The number of cores of virtual CPU, measureed in core.
- `memory` - The size of memory, measured in MB (Megabyte).
- `instance_type` - The type of instance.
- `charge_type` - The charge type of instance, possible values are: `year`, `month` and `dynamic` as pay by hour.
- `auto_renew` - Whether to renew an instance automatically or not.
- `remark` - The remarks of instance.
- `tag` - A tag assigned to the instance.
- `status` - Instance current status. Possible values are `Initializing`, `Starting`, `Running`, `Stopping`, `Stopped`, `Install Fail` and `Rebooting`.
- `create_time` - The time of creation for instance, formatted in RFC3339 time string.
- `expire_time` - The expiration time for instance, formatted in RFC3339 time string.
- `private_ip` - The private IP address assigned to the instance.
- `vpc_id` - The ID of VPC linked to the instance.
- `subnet_id` - The ID of subnet linked to the instance.
- `ip_set` - It is a nested type which documented below.
- `disk_set` - It is a nested type which documented below.

The attribute (`disk_set`) supports the following:

- `id` - The ID of disk.
- `size` - The size of disk, measured in GB (Gigabyte).
- `type` - The type of disk.
- `is_boot` - Specifies whether boot disk or not.

The attribute (`ip_set`) supports the following:

- `internet_type` - Type of Elastic IP routes.
- `ip` - Elastic IP address.

» ucloud_disks

This data source provides a list of Disk resources according to their Disk ID and disk type.

» Example Usage

```
data "ucloud_disks" "example" {}

output "first" {
  value = data.ucloud_disks.example.disks[0].id
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Optional) Availability zone where Disk are located. Such as: "cn-bj2-02". You may refer to list of availability zone
- `ids` - (Optional) A list of Disk IDs, all the Disks belong to this region will be retrieved if the ID is `[]`.
- `disk_type` - (Optional) The type of disk. Possible values are: `data_disk`as cloud disk, `ssd_data_disk` as SSD cloud disk, `system_disk`as system disk, `ssd_system_disk` as SSD system disk, `rssd_data_disk` as RDMA-SSD cloud disk.
- `name_regex` - (Optional) A regex string to filter resulting disks by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `disks` - It is a nested type which documented below.
- `total_count` - Total number of Disks that satisfy the condition.

The attribute (`disks`) support the following:

- `availability_zone` - Availability zone where disk is located.
- `id` - The ID of Disk.
- `name` - The name of Disk.
- `disk_size` - The size of disk. Purchase the size of disk in GB.
- `disk_type` - The type of disk.

- `charge_type` - The charge type of disk. Possible values are: `year` as pay by year, `month` as pay by month, `dynamic` as pay by hour.
- `tag` - A tag assigned to Disk.
- `create_time` - The creation time of Disk, formatted in RFC3339 time string.
- `expire_time` - The expiration time of disk, formatted in RFC3339 time string.
- `status` - The status of disk. Possible values are: `Available`, `InUse`, `Detaching`, `Initializating`, `Failed`, `Cloning`, `Restoring`, `RestoreFailed`.

» ucloud_instance

Provides an UHost Instance resource.

Note If you try to update some properties which requires stopping the instance, you must set `allow_stopping_for_update` to `true` in your config to allows Terraform to stop the instance to update its properties like `instance_type`, `root_password`, `boot_disk_size`, `data_disk_size`. In addition, once the instance complete creation, it takes around 10 minutes for boot disk initialization for the running instance, and the updates will only be made to some specific attributes (`root_password`, `boot_disk_size`) if required once the instance initialization completed.

» Example Usage

```
# Query default security group
data "ucloud_security_groups" "default" {
  type = "recommend_web"
}

# Query image
data "ucloud_images" "default" {
  availability_zone = "cn-bj2-04"
  name_regex        = "^CentOS 6.5 64"
  image_type        = "base"
}

# Create web instance
resource "ucloud_instance" "web" {
  availability_zone = "cn-bj2-04"
  image_id          = data.ucloud_images.default.images[0].id
  instance_type     = "n-basic-2"
  root_password     = "wA1234567"
```

```
  name              = "tf-example-instance"
  tag               = "tf-example"

  # the default Web Security Group that UCloud recommend to users
  security_group = data.ucloud_security_groups.default.security_groups[0].id
}

# Create cloud disk
resource "ucloud_disk" "example" {
  availability_zone = "cn-bj2-04"
  name              = "tf-example-instance"
  disk_size         = 30
}

# Attach cloud disk to instance
resource "ucloud_disk_attachment" "example" {
  availability_zone = "cn-bj2-04"
  disk_id           = ucloud_disk.example.id
  instance_id       = ucloud_instance.web.id
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Required, ForceNew) Availability zone where instance is located. such as: `cn-bj2-02`. You may refer to list of availability zone
- `image_id` - (Required) The ID for the image to use for the instance.
- `instance_type` - (Required) The type of instance, please visit the instance type table

  Note If you want to update this value, you must set `allow_stopping_for_update`to `true`.

- `allow_stopping_for_update` - (Optional) If you try to update some properties which requires stopping the instance, you must set `allow_stopping_for_update` to `true` in your config to allows Terraform to stop the instance to update its properties like `instance_type`, `root_password`, `boot_disk_size`, `data_disk_size`.
- `root_password` - (Optional) The password for the instance, which contains 8-30 characters, and at least 2 items of capital letters, lower case letters, numbers and special characters. The special characters include

`()~!@#$%&*-+=_|{}[]:;'<>,.?/`. If not specified, terraform will auto-generate a password.

Note If you want to update this value, you must set `allow_stopping_for_update` to `true`.

- `boot_disk_size` - (Optional) The size of the boot disk, measured in GB (GigaByte). Range: 20-100. The value set of disk size must be larger or equal to `20`(default: `20`) for Linux and `40` (default: `40`) for Windows. The responsive time is a bit longer if the value set is larger than default for local boot disk, and further settings may be required on host instance if the value set is larger than default for cloud boot disk. The disk volume adjustment must be a multiple of 10 GB. In addition, any reduction of boot disk size is not supported.

  Note If you want to update this value, you must set `allow_stopping_for_update` to `true`. In addition, when it is changed, you need to go to the instance for configuration.

- `boot_disk_type` - (Optional, ForceNew) The type of boot disk. Possible values are: `local_normal` and `local_ssd` for local boot disk, `cloud_ssd` for cloud SSD boot disk. (Default: `local_normal`). The `local_ssd` and `cloud_ssd` are not fully support by all regions as boot disk type, please proceed to UCloud console for more details.

- `data_disk_type` - (Optional, ForceNew) The type of local data disk. Possible values are: `local_normal` and `local_ssd` for local data disk. (Default: `local_normal`). The `local_ssd` is not fully support by all regions as data disk type, please proceed to UCloud console for more details. In addition, the `data_disk_type` must be same as `boot_disk_type` if specified.

- `data_disk_size` - (Optional) The size of local data disk, measured in GB (GigaByte), range: 0-8000 (Default: `20`), 0-8000 for cloud disk, 0-2000 for local sata disk and 100-1000 for local ssd disk (all the GPU type instances are included). The volume adjustment must be a multiple of 10 GB. In addition, any reduction of data disk size is not supported.

  Note If you want to update this value, you must set `allow_stopping_for_update` to `true`. In addition, when it is changed, you need to go to the instance for configuration.

- `charge_type` - (Optional, ForceNew) The charge type of instance, possible values are: `year`, `month` and `dynamic` as pay by hour (specific permission required). (Default: `month`).

- `duration` - (Optional, ForceNew) The duration that you will buy the instance (Default: `1`). The value is `0` when pay by month and the instance will be valid till the last day of that month. It is not required when `dynamic` (pay by hour).

- `name` - (Optional) The name of instance, which contains 1-63 characters and only support Chinese, English, numbers, '-', '_', '.'. If not specified, terraform will auto-generate a name beginning with `tf-instance`.
- `remark` - (Optional) The remarks of instance. (Default: `""`).
- `security_group` - (Optional) The ID of the associated security group.
- `vpc_id` - (Optional, ForceNew) The ID of VPC linked to the instance. If not defined `vpc_id`, the instance will use the default VPC in the current region.
- `subnet_id` - (Optional, ForceNew) The ID of subnet. If defined `vpc_id`, the `subnet_id` is Required. If not defined `vpc_id` and `subnet_id`, the instance will use the default subnet in the current region.
- `tag` - (Optional) A tag assigned to instance, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and '.'. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).
- `isolation_group` - (Optional, ForceNew) The ID of the associated isolation group.
- `private_ip` - (Optional, ForceNew) The private IP address assigned to the instance.
- `user_data` - (Optional, ForceNew) The user data to customize the startup behaviors when launching the instance. You may refer to user_data_document

» Attributes Reference

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

- `auto_renew` - Whether to renew an instance automatically or not.
- `cpu` - The number of cores of virtual CPU, measured in core.
- `memory` - The size of memory, measured in GB(Gigabyte).
- `create_time` - The time of creation for instance, formatted in RFC3339 time string.
- `expire_time` - The expiration time for instance, formatted in RFC3339 time string.
- `status` - Instance current status. Possible values are `Initializing`, `Starting`, `Running`, `Stopping`, `Stopped`, `Install Fail`, `ResizeFail` and `Rebooting`.
- `ip_set` - It is a nested type which documented below.
- `disk_set` - It is a nested type which documented below.

The attribute (`disk_set`) supports the following:

- `id` - The ID of disk.
- `size` - The size of disk, measured in GB (Gigabyte).
- `type` - The type of disk.
- `is_boot` - Specifies whether boot disk or not.

The attribute (`ip_set`) supports the following:

- `internet_type` - Type of Elastic IP routes. Possible values are: `International` as international BGP IP, `BGP` as china BGP IP and `Private` as private IP.
- `ip` - Elastic IP address.

» Import

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

```
$ terraform import ucloud_instance.example uhost-abcdefg
```

» ucloud_disk

Provides a Cloud Disk resource.

Note If the disk have attached to the instance, the instance will reboot automatically to make the change take effect when update the `disk_size`.

» Example Usage

```
# Query availability zone
data "ucloud_zones" "default" {}

# Create cloud disk
resource "ucloud_disk" "example" {
  availability_zone = data.ucloud_zones.default.zones[0].id
  name              = "tf-example-disk"
  disk_size         = 10
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Required, ForceNew) Availability zone where cloud disk is located. Such as: "cn-bj2-02". You may refer to list of availability zone.
- `disk_size` - (Required) The size of disk. Purchase the size of disk in GB. 1-8000 for a cloud disk, 1-4000 for SSD cloud disk. If the disk have attached to the instance, the instance will reboot automatically to make the change take effect when update the `disk_size`.

- `name` - (Optional) The name of disk, should have 6-63 characters and only support Chinese, English, numbers, '-', '_'. If not specified, terraform will auto-generate a name beginning with `tf-disk`.
- `disk_type` - (Optional, ForceNew) The type of disk. Possible values are: `data_disk`as cloud disk, `ssd_data_disk` as ssd cloud disk, `rssd_data_disk` as RDMA-SSD cloud disk (the `rssd_data_disk` only be supported in `cn-bj2-05`).(Default: `data_disk`).
- `charge_type` - (Optional, ForceNew) Charge type of disk. Possible values are: `year` as pay by year, `month` as pay by month, `dynamic` as pay by hour. (Default: `month`).
- `duration` - (Optional, ForceNew) The duration that you will buy the resource. (Default: `1`). It is not required when `dynamic` (pay by hour), the value is `0` when `month`(pay by month) and the disk will be vaild till the last day of that month.
- `tag` - (Optional, ForceNew) A tag assigned to VPC, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and ''. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).

» Attributes Reference

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

- `create_time` - The time of creation of disk, formatted in RFC3339 time string.
- `expire_time` - The expiration time of disk, formatted in RFC3339 time string.
- `status` - The status of disk. Possible values are: `Available`, `InUse`, `Detaching`, `Initializating`, `Failed`, `Cloning`, `Restoring`, `RestoreFailed`.

» Import

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

```
$ terraform import ucloud_disk.example bsm-abcdefg
```

» ucloud_disk_attachment

Provides a Cloud Disk Attachment resource for attaching Cloud Disk to UHost Instance.

» Example Usage

```
# Query availability zone
data "ucloud_zones" "default" {}

# Query image
data "ucloud_images" "default" {
  availability_zone = data.ucloud_zones.default.zones[0].id
  name_regex        = "^CentOS 7.[1-2] 64"
  image_type        = "base"
}

# Create cloud disk
resource "ucloud_disk" "default" {
  availability_zone = data.ucloud_zones.default.zones[0].id
  name              = "tf-example-disk"
  disk_size         = 10
}

# Create a web server
resource "ucloud_instance" "web" {
  availability_zone = data.ucloud_zones.default.zones[0].id
  instance_type     = "n-basic-2"

  image_id      = data.ucloud_images.default.images[0].id
  root_password = "wA1234567"

  name = "tf-example-disk"
  tag  = "tf-example"
}

# attach cloud disk to instance
resource "ucloud_disk_attachment" "default" {
  availability_zone = data.ucloud_zones.default.zones[0].id
  disk_id           = ucloud_disk.default.id
  instance_id       = ucloud_instance.web.id
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Required, ForceNew) The Zone to attach the disk in.
- `instance_id` - (Required, ForceNew) The ID of host instance.
- `disk_id` - (Required, ForceNew) The ID of disk that needs to be attached

» ucloud_isolation_group

Provides an Isolation Group resource. The Isolation Group is a logical group of UHost instance, which ensure that each UHost instance within a group is on a different physical machine. Up to seven UHost instance can be added per isolation group in a single availability_zone.

» Example Usage

```
resource "ucloud_isolation_group" "foo" {
  name   = "tf-acc-isolation-group"
  remark = "test"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Optional, ForceNew) The name of the isolation group information which contains 1-63 characters and only support Chinese, English, numbers, '-', '_', '', ',', '[', ']', ':'. If not specified, terraform will auto-generate a name beginning with `tf-isolation-group`.
- `remark` - (Optional, ForceNew) The remarks of the isolation group. (Default: `""`).

» Import

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

```
$ terraform import ucloud_isolation_group.example ig-abc123456
```

» ucloud_eips

This data source provides a list of EIP resources (Elastic IP address) according to their EIP ID.

» Example Usage

```
data "ucloud_eips" "example" {}

output "first" {
  value = data.ucloud_eips.example.eips[0].ip_set[0].ip
}
```

» Argument Reference

The following arguments are supported:

- `ids` - (Optional) A list of Elastic IP IDs, all the EIPs belong to this region will be retrieved if the ID is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting eips by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `eips` - It is a nested type which documented below.
- `total_count` - Total number of Elastic IPs that satisfy the condition.

The attribute (`eips`) support the following:

- `bandwidth` - Maximum bandwidth to the elastic public network, measured in Mbps.
- `ip_set` - It is a nested type which documented below.
- `create_time` - The creation time of Elastic IP, formatted in RFC3339 time string.
- `expire_time` - The expiration time for Elastic IP, formatted in RFC3339 time string.
- `charge_mode` - The charge mode of Elastic IP. Possible values are: `traffic` as pay by traffic, `bandwidth` as pay by bandwidth.
- `charge_type` - The charge type of Elastic IP. Possible values are: `year` as pay by year, `month` as pay by month, `dynamic` as pay by hour.

- `name` - The name of Elastic IP.
- `remark` - The remarks of Elastic IP.
- `status` - Elastic IP status. Possible values are: `used` as in use, `free` as available and `freeze` as associating.
- `tag` - A tag assigned to Elastic IP.

The attribute (`ip_set`) support the following:

- `internet_type` - Type of Elastic IP routes.
- `ip` - Elastic IP address.

» ucloud_security_groups

This data source provides a list of Security Group resources according to their Security Group ID, name and resource id.

» Example Usage

```
data "ucloud_security_groups" "example" {}

output "first" {
  value = data.ucloud_security_groups.example.security_groups[0].id
}
```

» Argument Reference

The following arguments are supported:

- `ids` - (Optional) A list of Security Group IDs, all the Security Group resources belong to this region will be retrieved if the ID is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting Security Group resources by name.
- `type` - (Optional) The type of Security Group. Possible values are: `recommend_web` as the default Web security group that UCloud recommend to users, default opened port include 80, 443, 22, 3389, `recommend_non_web` as the default non Web security group that UCloud recommend to users, default opened port include 22, 3389, `user_defined` as the security groups defined by users. You may refer to security group.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `security_groups` - It is a nested type which documented below.
- `total_count` - Total number of Security Group resources that satisfy the condition.

The attribute (`security_groups`) support the following:

- `id` - The ID of Security Group.
- `name` - The name of Security Group.
- `rules` - It is a nested type which documented below.
- `type` - The type of Security Group.
- `remark` - The remarks of the security group.
- `tag` - A tag assigned to the security group.
- `create_time` - The time of creation for the security group, formatted in RFC3339 time string.

The attribute (`rules`) support the following:

- `cidr_block` - The cidr block of source.
- `policy` - Authorization policy. Can be either `accept` or `drop`.
- `port_range` - The range of port numbers, range: 1-65535. (eg: `port` or `port1-port2`).
- `priority` - Rule priority. Can be `high`, `medium`, `low`.
- `protocol` - The protocol. Can be `tcp`, `udp`, `icmp`, `gre`.

» ucloud_eip

Provides an Elastic IP resource.

» Example Usage

```
resource "ucloud_eip" "example" {
  bandwidth     = 2
  charge_mode   = "bandwidth"
  name          = "tf-example-eip"
  tag           = "tf-example"
  internet_type = "bgp"
}
```

» Argument Reference

The following arguments are supported:

- `internet_type` - (Required, ForceNew) Type of Elastic IP routes. Possible values are: `international` as international BGP IP and `bgp` as china mainland BGP IP.

- `bandwidth` - (Optional) Maximum bandwidth to the elastic public network, measured in Mbps (Mega bit per second). The ranges for bandwidth are: 1-200 for pay by traffic, 1-800 for pay by bandwidth. (Default: `1`).
- `share_bandwidth_package_id` - (Optional) The Id of Share Bandwidth Package. If it is filled in, the `charge_mode` can only be set with `share_bandwidth`.
- `duration` - (Optional, ForceNew) The duration that you will buy the resource. (Default: `1`). It is not required when `dynamic` (pay by hour), the value is `0` when `month`(pay by month) and the instance will be valid till the last day of that month.
- `charge_mode` -(Optional) Elastic IP charge mode. Possible values are: `traffic` as pay by traffic, `bandwidth` as pay by bandwidth, `share_bandwidth` as share bandwidth mode. (Default: `bandwidth`for the Elastic IP, `share_bandwidth` for the Elastic IP with share bandwidth mode).
- `charge_type` - (Optional, ForceNew) Elastic IP charge type. Possible values are: `year` as pay by year, `month` as pay by month, `dynamic` as pay by hour (specific permission required). (Default: `month`).
- `name` - (Optional) The name of the EIP, which contains 1-63 characters and only support Chinese, English, numbers, '-', '_', '.'. If not specified, terraform will auto-generate a name beginning with `tf-eip`.
- `remark` - (Optional) The remarks of the EIP. (Default: `""`).
- `tag` - (Optional) A tag assigned to Elastic IP, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and '.'. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).

» Attributes Reference

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

- `create_time` - The time of creation for EIP, formatted in RFC3339 time string.
- `expire_time` - The expiration time for EIP, formatted in RFC3339 time string.
- `ip_set` - It is a nested type which documented below.
- `resource` - It is a nested type which documented below.

- `status` - EIP status. Possible values are: `used` as in use, `free` as available and `freeze` as associating.
- `public_ip` - Public IP address of Elastic IP.

The attribute (`ip_set`) support the following:

- `internet_type` - Type of Elastic IP routes.

The attribute (`resource`) support the following:

- `id` - The ID of the resource with EIP attached.
- `type` - The type of resource with EIP attached. Possible values are `instance` as instance, `lb` as load balancer.

» Import

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

```
$ terraform import ucloud_eip.example eip-abcdefg
```

» ucloud_eip_association

Provides an EIP Association resource for associating Elastic IP to UHost Instance, Load Balancer, etc.

» Example Usage

```
# Query availability zone
data "ucloud_zones" "default" {}

# Query image
data "ucloud_images" "default" {
  availability_zone = data.ucloud_zones.default.zones[0].id
  name_regex        = "^CentOS 7.[1-2] 64"
  image_type        = "base"
}

# Create security group
resource "ucloud_security_group" "default" {
  name = "tf-example-eip"
  tag  = "tf-example"

  rules {
```

```
    port_range = "80"
    protocol   = "tcp"
    cidr_block = "0.0.0.0/0"
    policy     = "accept"
  }
}

# Create an eip
resource "ucloud_eip" "default" {
  bandwidth     = 2
  charge_mode   = "bandwidth"
  name          = "tf-example-eip"
  tag           = "tf-example"
  internet_type = "bgp"
}

# Create a web server
resource "ucloud_instance" "web" {
  instance_type     = "n-basic-2"
  availability_zone = data.ucloud_zones.default.zones[0].id
  image_id          = data.ucloud_images.default.images[0].id

  data_disk_size = 50
  root_password  = "wA1234567"
  security_group = ucloud_security_group.default.id

  name = "tf-example-eip"
  tag  = "tf-example"
}

# Bind eip to instance
resource "ucloud_eip_association" "default" {
  resource_id = ucloud_instance.web.id
  eip_id      = ucloud_eip.default.id
}
```

» Argument Reference

The following arguments are supported:

- `eip_id` - (Required, ForceNew) The ID of EIP.
- `resource_id` - (Required, ForceNew) The ID of resource with EIP attached.
- `resource_type` - (**Deprecated**, ForceNew), attribute `resource_type` is deprecated for optimizing parameters.

» ucloud_security_group

Provides a Security Group resource.

» Example Usage

```
resource "ucloud_security_group" "example" {
  name = "tf-example-security-group"
  tag  = "tf-example"

  # http access from LAN
  rules {
    port_range = "80"
    protocol   = "tcp"
    cidr_block = "192.168.0.0/16"
    policy     = "accept"
  }

  # https access from LAN
  rules {
    port_range = "443"
    protocol   = "tcp"
    cidr_block = "192.168.0.0/16"
    policy     = "accept"
  }
}
```

» Argument Reference

The following arguments are supported:

- `rules` - (Required) A list of security group rules. Can be specified multiple times for each rules. Each rules supports fields documented below.

- `name` - (Optional) The name of the security group which contains 1-63 characters and only support Chinese, English, numbers, '-', '_' and '.'. If not specified, terraform will auto-generate a name beginning with `tf-security-group`.
- `remark` - (Optional) The remarks of the security group. (Default: `""`).
- `tag` - (Optional) A tag assigned to security group, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and '.'. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).

» Block rules

The rules mapping supports the following:

- `port_range` - (Optional) The range of port numbers, range: 1-65535. (eg: `port` or `port1-port2`).
- `cidr_block` - (Optional) The cidr block of source.
- `policy` - (Optional) Authorization policy. Possible values are: `accept`, `drop`.
- `priority` - (Optional) Rule priority. Possible values are: `high`, `medium`, `low`.
- `protocol` - (Optional) The protocol. Possible values are: `tcp`, `udp`, `icmp`, `gre`.

» Attributes Reference

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

- `create_time` - The time of creation of security group, formatted in RFC3339 time string.

» Import

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

```
$ terraform import ucloud_security_group.example firewall-abc123456
```

» ucloud_lbs

This data source provides a list of Load Balancer resources according to their Load Balancer ID, VPC ID and Subnet ID.

» Example Usage

```
data "ucloud_lbs" "example" {
}

output "first" {
  value = data.ucloud_lbs.example.lbs[0].id
}
```

» Argument Reference

The following arguments are supported:

- `ids` - (Optional) A list of Load Balancer IDs, all the LBs belong to this region will be retrieved if the ID is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting lbs by name.
- `vpc_id` - (Optional) The ID of the VPC linked to the Load Balancers.
- `subnet_id` - (Optional) The ID of subnet that intrant load balancer belongs to.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `lbs` - It is a nested type which documented below.
- `total_count` - Total number of Load Balancers that satisfy the condition.

The attribute (`lbs`) support the following:

- `id` - The ID of Load Balancer.
- `name` - The name of Load Balancer.
- `internal` - Indicate whether the load balancer is intranet.
- `tag` - A tag assigned to Load Balancer.
- `remark` - The remarks of Load Balancer.
- `vpc_id` - The ID of the VPC linked to the Load Balancers.
- `subnet_id` - (Optional) The ID of subnet that intrant load balancer belongs to.
- `private_ip` - The IP address of intranet IP.
- `create_time` - The creation time of Load Balancer, formatted in RFC3339 time string.

The attribute (`ip_set`) support the following:

- `internet_type` - Type of Load Balancer routes.
- `ip` - Load Balancer address.

» ucloud_lb_listeners

This data source provides a list of Load Balancer Listener resources according to their Load Balancer Listener ID.

» Example Usage

```
data "ucloud_lb_listeners" "example" {
  load_balancer_id = "ulb-xxx"
}

output "first" {
  value = data.ucloud_lb_listeners.example.lb_listeners[0].id
}
```

» Argument Reference

The following arguments are supported:

- `load_balancer_id` - (Required) The ID of a load balancer.

- `ids` - (Optional) A list of LB Listener IDs, all the LB Listeners belong to this region will be retrieved if the ID is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting lb listeners by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `lb_listeners` - It is a nested type which documented below.
- `total_count` - Total number of LB listeners that satisfy the condition.

The attribute (`lb_listeners`) support the following:

- `id` - The ID of LB Listener.
- `name` - The name of LB Listener.
- `protocol` - LB Listener protocol. Possible values: `http`, `https` if `listen_type` is `request_proxy`, `tcp` and `udp` if `listen_type` is `packets_transmit`.
- `listen_type` - The type of LB Listener. Possible values are `request_proxy` and `packets_transmit`.
- `port` - Port opened on the LB Listener to receive requests, range: 1-65535.
- `idle_timeout` - Amount of time in seconds to wait for the response for in between two sessions if `listen_type` is `request_proxy`, range: 0-86400. Amount of time in seconds to wait for one session if `listen_type` is

`packets_transmit`, range: 60-900. The session will be closed as soon as no response if it is `0`.

- `method` - The load balancer method in which the listener is. Possible values are: `roundrobin`, `source`, `consistent_hash`, `source_port` , `consistent_hash_port`, `weight_roundrobin` and `leastconn`.
  - The `consistent_hash`, `source_port` , `consistent_hash_port`, `roundrobin`, `source` and `weight_roundrobin` are valid if `listen_type` is `packets_transmit`.
  - The `rundrobin`, `source` and `weight_roundrobin` and `leastconn` are vaild if `listen_type` is `request_proxy`.
- `persistence` - Indicate whether the persistence session is enabled, it is invaild if `persistence_type` is `none`, an auto-generated string will be exported if `persistence_type` is `server_insert`, a custom string will be exported if `persistence_type` is `user_defined`.
- `persistence_type` - The type of session persistence of LB Listener. Possible values are: `none` as disabled, `server_insert` as auto-generated string and `user_defined` as cutom string. (Default: `none`).
- `health_check_type` - Health check method. Possible values are `port` as port checking and `path` as http checking.
- `path` - Health check path checking.
- `domain` - Health check domain checking.
- `status` - LB Listener status. Possible values are: `allNormal` for all resource functioning well, `partNormal` for partial resource functioning well and `allException` for all resource functioning exceptional.

» ucloud_lb_attachments

This data source provides a list of Load Balancer Attachment resources according to their Load Balancer Attachment ID.

» Example Usage

```
data "ucloud_lb_attachments" "example" {
  load_balancer_id = "ulb-xxx"
  listener_id      = "vserver-xxx"
}

output "first" {
  value = data.ucloud_lb_attachments.example.lb_attachments[0].id
}
```

» Argument Reference

The following arguments are supported:

- `load_balancer_id` - (Required) The ID of a load balancer.
- `listener_id` - (Required) The ID of a listener server.
- `ids` - (Optional) A list of LB Attachment IDs, all the LB Attachments belong to the Load Balancer listener will be retrieved if the ID is `[]`.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `lb_attachments` - It is a nested type which documented below.
- `total_count` - Total number of LB Attachments that satisfy the condition.

The attribute (`lb_attachments`) support the following:

- `id` - The ID of LB Attachment.
- `resource_id` - The ID of a backend server.
- `port` - Port opened on the backend server to receive requests, range: 1-65535.
- `private_ip` - The private ip address for backend servers.
- `status` - The status of backend servers. Possible values are: `normalRunning`, `exceptionRunning`.

» ucloud_lb_rules

This data source provides a list of Load Balancer Rule resources according to their Load Balancer Rule ID.

» Example Usage

```
data "ucloud_lb_rules" "example" {
  load_balancer_id = "ulb-xxx"
  listener_id      = "vserver-xxx"
}

output "first" {
  value = data.ucloud_lb_rules.example.lb_rules[0].id
}
```

» Argument Reference

The following arguments are supported:

- `load_balancer_id` - (Required) The ID of a load balancer.
- `listener_id` - (Required) The ID of a listener server.

- `ids` - (Optional) A list of LB Rule IDs, all the LB Rules belong to the Load Balancer listener will be retrieved if the ID is `[]`.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `lb_rules` - It is a nested type which documented below.
- `total_count` - Total number of LB Rules that satisfy the condition.

The attribute (`lb_rules`) support the following:

- `id` - The ID of LB Rule.
- `path` - (Optional) The path of Content forward matching fields. `path` and `domain` cannot coexist.
- `domain` - (Optional) The domain of content forward matching fields. `path` and `domain` cannot coexist.

» ucloud_lb_ssls

This data source provides a list of Load Balancer SSL certificate resources according to their Load Balancer SSL certificate resource ID and name.

» Example Usage

```
data "ucloud_lb_ssls" "example" {
}

output "first" {
  value = data.ucloud_lb_ssls.example.lb_ssls[0].id
}
```

» Argument Reference

The following arguments are supported:

- `ids` - (Optional) A list of LB SSL certificate resource IDs, all the LB SSL certificate resources in the current region will be retrieved if the ID is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting LB SSL by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `lb_ssls` - It is a nested type which documented below.
- `total_count` - Total number of LB SSL certificate resources that satisfy the condition.

The attribute (`lb_ssls`) support the following:

- `id` - The ID of LB SSL certificate resource.
- `name` - The name of LB SSL certificate resource.
- `create_time` - The time of creation for lb ssl, formatted in RFC3339 time string.

» ucloud_lb

Provides a Load Balancer resource.

» Example Usage

```
resource "ucloud_lb" "web" {
  name = "tf-example-lb"
  tag  = "tf-example"
}
```

» Argument Reference

The following arguments are supported:

- `internal` - (Optional, ForceNew) Indicate whether the load balancer is intranet mode.(Default: `"false"`)

- `name` - (Optional) The name of the load balancer. If not specified, terraform will auto-generate a name beginning with `tf-lb`.
- `charge_type` - (**Deprecated**, ForceNew), argument `charge_type` is deprecated for optimizing parameters.
- `vpc_id` - (Optional, ForceNew) The ID of the VPC linked to the Load balancer, This argument is not required if default VPC.
- `subnet_id` - (Optional, ForceNew) The ID of subnet that intranet load balancer belongs to. This argument is not required if default subnet.
- `tag` - (Optional) A tag assigned to load balancer, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and '.'. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).
- `remark` - (Optional) The remarks of the load balancer. (Default: `""`).
- `security_group` - (Optional) The ID of the associated security group. The security_group only takes effect for ULB instances of request_proxy mode and extranet mode at present.

» Attributes Reference

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

- `create_time` - The time of creation for load balancer, formatted in RFC3339 time string.
- `expire_time` - **Deprecated** attribute `expire_time` is deprecated for optimizing outputs.
- `ip_set` - It is a nested type which documented below.
- `private_ip` - The IP address of intranet IP. It is `""` if `internal` is `false`.

The attribute (`ip_set`) support the following:

- `internet_type` - Type of Elastic IP routes.
- `ip` - Elastic IP address.

» Import

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

```
$ terraform import ucloud_lb.example ulb-abc123456
```

» ucloud_lb_attachment

Provides a Load Balancer Attachment resource for attaching Load Balancer to UHost Instance, etc.

» Example Usage

```
# Query image
data "ucloud_images" "default" {
  availability_zone = "cn-bj2-04"
  name_regex        = "^CentOS 6.5 64"
  image_type        = "base"
}

# Create Load Balancer
resource "ucloud_lb" "web" {
  name = "tf-example-lb"
  tag  = "tf-example"
}

# Create Load Balancer Listener with http protocol
resource "ucloud_lb_listener" "default" {
  load_balancer_id = ucloud_lb.web.id
  protocol         = "http"
}

# Create web server
resource "ucloud_instance" "web" {
  instance_type     = "n-basic-2"
  availability_zone = "cn-bj2-04"

  root_password = "wA1234567"
  image_id      = data.ucloud_images.default.images[0].id

  name = "tf-example-lb"
  tag  = "tf-example"
}

# Attach instances to Load Balancer
resource "ucloud_lb_attachment" "example" {
  load_balancer_id = ucloud_lb.web.id
  listener_id      = ucloud_lb_listener.default.id
  resource_id      = ucloud_instance.web.id
  port             = 80
}
```

» Argument Reference

The following arguments are supported:

- `load_balancer_id` - (Required, ForceNew) The ID of a load balancer.
- `listener_id` - (Required, ForceNew) The ID of a listener server.
- `resource_id` - (Required, ForceNew) The ID of a backend server.

- `resource_type` - (**Deprecated**, ForceNew), attribute `resource_type` is deprecated for optimizing parameters.
- `port` - (Optional) The listening port of the backend server, range: 1-65535, (Default: `80`). Backend server port have the following restrictions: If the LB listener type is `request_proxy`, the backend serve can add different ports to implement different service instances of the same IP. Else if LB listener type is `packets_transmit`, the port of the backend server must be consistent with the LB listening port.

» Attributes Reference

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

- `private_ip` - The private ip address for backend servers.
- `status` - The status of backend servers. Possible values are: `normalRunning`, `exceptionRunning`.

» ucloud_lb_listener

Provides a Load Balancer Listener resource.

Note This `listen_type` only support when `protocol` is `tcp` in the extranet mode and the default value is `request_proxy`. In addition, in the extranet mode, the `listen_type` is `request_proxy` if `protocol`is `http` or `https`, the `listen_type` is `packets_transmit` if `protocol`is `udp`. In the intranet mode, the `listen_type` is `packets_transmit`.

» Example Usage

```
resource "ucloud_lb" "web" {
  name = "tf-example-lb"
  tag  = "tf-example"
}

resource "ucloud_lb_listener" "example" {
  load_balancer_id = ucloud_lb.web.id
  protocol         = "http"
}
```

» Argument Reference

The following arguments are supported:

- `load_balancer_id` - (Required, ForceNew) The ID of load balancer instance.
- `protocol` - (Required, ForceNew) Listener protocol. Possible values: `http`, `https`, `tcp` if `listen_type` is `request_proxy`, `tcp` and `udp` if `listen_type` is `packets_transmit`.

- `name` - (Optional) The name of the listener. If not specified, terraform will auto-generate a name beginning with `tf-lb-listener`.
- `listen_type` - (Optional, ForceNew) The type of listener. Possible values are `request_proxy` and `packets_transmit`. When `packets_transmit` was specified, you need to config the instances by yourself if the instances attach to the load balancer. You may refer to configuration instruction.
- `port` - (Optional, ForceNew) Port opened on the listeners to receive requests, range: 1-65535. The default value: `80` as `protocol` is `http`, `443` as `protocol` is `https`, `1024` as `protocol` is `tcp` or `udp`.
- `idle_timeout` - (Optional) Keep alive timeout of the connection between the client and LB, measured in second. Range: 0-86400 when `listen_type` is `request_proxy`, range: 60-900 when `listen_type` is `packets_transmit` (Default: `60`). The connection will be closed as soon as no response between the client and LB if it set by `0`.
- `method` - (Optional) The load balancer method in which the listener is. Possible values are: `roundrobin`, `source`, `consistent_hash`, `source_port` , `consistent_hash_port`, `weight_roundrobin` and `leastconn`. (Default: `roundrobin`).
  - The `consistent_hash`, `source_port` , `consistent_hash_port`, `roundrobin`, `source` and `weight_roundrobin` are valid if `listen_type` is `packets_transmit`.
  - The `roundrobin`, `source` and `weight_roundrobin` and `leastconn` are valid if `listen_type` is `request_proxy`.
- `persistence` - (Optional) Indicate whether the persistence session is enabled, it is invalid if `persistence_type` is `none`, an auto-generated string will be exported if `persistence_type` is `server_insert`, a custom string will be exported if `persistence_type` is `user_defined`.
- `persistence_type` - (Optional) The type of session persistence of listener. Possible values are: `none` as disabled, `server_insert` as auto-generated key and `user_defined` as customized key. (Default: `none`).
- `health_check_type` - (Optional) Health check method. Possible values are `port` as port checking and `path` as http checking.
- `path` - (Optional) Health check path checking.
- `domain` - (Optional) Health check domain checking.

» Attributes Reference

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

- `status` - Listener status. Possible values are: `allNormal` for all resource functioning well, `partNormal` for partial resource functioning well and `allException` for all resource functioning exceptional.

» Import

LB Listener can be imported using the `id`, e.g.

```
$ terraform import ucloud_lb_listener.example vserver-abcdefg
```

» ucloud_lb_rule

Provides a Load Balancer Rule resource to add content forwarding policies for Load Balancer backend resource.

Note The Load Balancer Rule can only be define while the `protocol` of lb listener is one of HTTP and HTTPS. In addition, should set one of `domain` and `path` if defined.

» Example Usage

```
data "ucloud_images" "default" {
  availability_zone = "cn-bj2-02"
  name_regex        = "^CentOS 6.5 64"
  image_type        = "base"
}

resource "ucloud_lb" "web" {
  name = "tf-example-lb"
  tag  = "tf-example"
}

resource "ucloud_lb_listener" "default" {
  load_balancer_id = ucloud_lb.web.id
  protocol         = "http"
}

resource "ucloud_instance" "web" {
  instance_type     = "n-basic-2"
  availability_zone = "cn-bj2-02"
```

```
  root_password = "wA1234567"
  image_id      = data.ucloud_images.default.images[0].id

  name = "tf-example-lb"
  tag  = "tf-example"
}

resource "ucloud_lb_attachment" "default" {
  load_balancer_id = ucloud_lb.web.id
  listener_id      = ucloud_lb_listener.default.id
  resource_type    = "instance"
  resource_id      = ucloud_instance.web.id
  port             = 80
}

resource "ucloud_lb_rule" "example" {
  load_balancer_id = ucloud_lb.web.id
  listener_id      = ucloud_lb_listener.default.id
  backend_ids      = ucloud_lb_attachment.default.*.id
  domain           = "www.ucloud.cn"
}
```

» Argument Reference

The following arguments are supported:

- `load_balancer_id` - (Required, ForceNew) The ID of a load balancer.
- `listener_id` - (Required, ForceNew) The ID of a listener server.
- `backend_ids` - (Required, ForceNew) The IDs of the backend servers where rule applies, this argument is populated base on the `backend_id` responded from `lb_attachment` create.

- `path` - (Optional) The path of Content forward matching fields. `path` and `domain` cannot coexist. `path` and `domain` must be filled in one.
- `domain` - (Optional) The domain of content forward matching fields. `path` and `domain` cannot coexist. `path` and `domain` must be filled in one.

» ucloud_lb_ssl

Provides a Load Balancer SSL certificate resource.

» Example Usage

```
resource "ucloud_lb_ssl" "default" {
  name        = "tf-example-lb-ssl"
  private_key = file("private.key")
  user_cert   = file("user.crt")
  ca_cert     = file("ca.crt")
}
```

» Argument Reference

The following arguments are supported:

- `private_key` - (Required, ForceNew) The content of the private key about ssl certificate.
- `user_cert` - (Required, ForceNew) The content of the user certificate about ssl certificate.

- `name` - (Optional, ForceNew) The name of the LB ssl, which contains 1-63 characters and only support Chinese, English, numbers, '-', '_', '.'. If not specified, terraform will auto-generate a name beginning with `tf-lb-ssl`.
- `ca_cert` - (Optional, ForceNew) The content of the CA certificate about ssl certificate.

» Attributes Reference

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

- `create_time` - The time of creation for lb ssl, formatted in RFC3339 time string.

» ucloud_lb_ssl_attachment

Provides a Load Balancer SSL attachment resource for attaching SSL certificate to Load Balancer Listener.

» Example Usage

```
resource "ucloud_lb" "foo" {
  name = "tf-example-lb-ssl-attachment"
  tag  = "tf-example"
```

```
}

resource "ucloud_lb_listener" "foo" {
  name             = "tf-example-lb-ssl-attachment"
  load_balancer_id = ucloud_lb.foo.id
  protocol         = "https"
}

resource "ucloud_lb_ssl" "foo" {
  name        = "tf-example-lb-ssl-attachment"
  private_key = file("private.key")
  user_cert   = file("user.crt")
  ca_cert     = file("ca.crt")
}


resource "ucloud_lb_ssl_attachment" "foo" {
  load_balancer_id = ucloud_lb.foo.id
  listener_id      = ucloud_lb_listener.foo.id
  ssl_id           = ucloud_lb_ssl.foo.id
}
```

» Argument Reference

The following arguments are supported:

- `ssl_id` - (Required, ForceNew) The ID of SSL certificate.
- `load_balance_id` - (Required, ForceNew) The ID of load balancer instance.
- `listener_id` - (Required, ForceNew) The ID of listener servers.

» ucloud_vpcs

This data source provides a list of VPC resources according to their VPC ID, name.

» Example Usage

```
data "ucloud_vpcs" "example" {
}

output "first" {
  value = data.ucloud_vpcs.example.vpcs[0].id
```

```
}
```

» Argument Reference

The following arguments are supported:

- `ids` - (Optional) A list of VPC IDs, all the VPC resources belong to this region will be retrieved if the ID is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting VPC resources by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `vpcs` - It is a nested type which documented below.
- `total_count` - Total number of VPC resources that satisfy the condition.

The attribute (`vpcs`) support the following:

- `id` - The ID of VPC.
- `name` - The name of VPC.
- `cidr_blocks` - The CIDR blocks of VPC.
- `tag` - A tag assigned to VPC.
- `create_time` - The time of creation for VPC, formatted in RFC3339 time string.
- `update_time` - The time whenever there is a change made to VPC, formatted in RFC3339 time string.

» ucloud_subnets

This data source provides a list of Subnet resources according to their Subnet ID, name and the VPC they belong to.

» Example Usage

```
data "ucloud_subnets" "example" {
  vpc_id = "uvnet-xxx"
}
```

```
output "first" {
  value = data.ucloud_subnets.example.subnets[0].id
}
```

» Argument Reference

The following arguments are supported:

- `ids` - (Optional) A list of Subnet IDs, all the Subnet resources belong to this region will be retrieved if the ID is `[]`.
- `vpc_id` - (Optional) The id of the VPC that the desired Subnet belongs to.
- `name_regex` - (Optional) A regex string to filter resulting Subnet resources by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `subnets` - It is a nested type which documented below.
- `total_count` - Total number of Subnet resources that satisfy the condition.

The attribute (`subnets`) support the following:

- `id` - The ID of Subnet.
- `name` - The name of Subnet.
- `cidr_block` - The cidr block of the desired Subnet.
- `create_time` - The time of creation of Subnet, formatted in RFC3339 time string.
- `remark` - The remark of the Subnet.
- `tag` - A tag assigned to Subnet.

» ucloud_nat_gateways

This data source providers a list of Nat Gateway resources according to their ID and name.

» Example Usage

```
data "ucloud_nat_gateways" "example" {
}

output "first" {
  value = data.ucloud_nat_gateways.example.nat_gateways[0].id
}
```

» Argument Reference

The following arguments are supported:

- `ids` - (Optional) A list of Nat Gateway IDs, all the Nat Gateways belongs to the defined region will be retrieved if this argument is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting Nat Gateways by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `nat_gateways` - It is a nested type. Nat Gateways documented below.
- `total_count` - Total number of Nat Gateways that satisfy the condition.

The attribute (`nat_gateways`) support the following:

- `id` - The ID of Nat Gateway.
- `name` - The name of the Nat Gateway.
- `remark` - The remarks of Nat Gateway.
- `tag` - A tag assigned to the Nat Gateway.
- `vpc_id` - The ID of VPC linked to the Nat Gateway.
- `subnet_ids` - The list of subnet ID under the VPC.
- `security_group` -The ID of the associated security group.
- `create_time` - The time of creation for Nat Gateway, formatted in RFC3339 time string.
- `ip_set` - It is a nested type which documented below.

The attribute (`ip_set`) supports the following:

- `internet_type` - Type of Elastic IP routes.
- `ip` - Elastic IP address.

» ucloud_vpc

Provides a VPC resource.

Note The network segment can only be created or deleted, can not perform both of them at the same time.

» Example Usage

```
resource "ucloud_vpc" "example" {
  name = "tf-example-vpc"
  tag  = "tf-example"

  # vpc network
  cidr_blocks = ["192.168.0.0/16"]
}
```

» Argument Reference

The following arguments are supported:

- `cidr_blocks` - (Required) The CIDR blocks of VPC.

- `name` - (Optional, ForceNew) The name of VPC. If not specified, terraform will auto-generate a name beginning with `tf-vpc`.
- `tag` - (Optional, ForceNew) A tag assigned to VPC, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and '.'. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).
- `remark` - (Optional, ForceNew) The remarks of the VPC. (Default: `""`).

» Attributes Reference

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

- `create_time` - The time of creation for VPC, formatted in RFC3339 time string.
- `update_time` - The time whenever there is a change made to VPC, formatted in RFC3339 time string.
- `network_info` - It is a nested type which documented below.

The attribute (`network_info`) support the following:

- `cidr_block` - The CIDR block of the VPC.

» Import

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

```
$ terraform import ucloud_vpc.example uvnet-abc123456
```

» ucloud_vpc_peering_connection

Provides an VPC Peering Connection for establishing a connection between multiple VPC.

» Example Usage

```
resource "ucloud_vpc" "foo" {
  name        = "tf-example-vpc-01"
  tag         = "tf-example"
  cidr_blocks = ["192.168.0.0/16"]
}

resource "ucloud_vpc" "bar" {
  name        = "tf-example-vpc-02"
  tag         = "tf-example"
  cidr_blocks = ["10.10.0.0/16"]
}

resource "ucloud_vpc_peering_connection" "connection" {
  vpc_id      = ucloud_vpc.foo.id
  peer_vpc_id = ucloud_vpc.bar.id
}
```

» Argument Reference

The following arguments are supported:

- `vpc_id` - (Required, ForceNew) The short of ID of the requester VPC of the specific VPC Peering Connection to retrieve.
- `peer_vpc_id` - (Required, ForceNew) The short ID of accepter VPC of the specific VPC Peering Connection to retrieve.

- `peer_project_id` - (Optional, ForceNew) The ID of accepter project of the specific VPC Peering Connection to retrieve.

» ucloud_subnet

Provides a Subnet resource under VPC resource.

» Example Usage

```
resource "ucloud_vpc" "default" {
  name = "tf-example-vpc"
  tag  = "tf-example"

  # vpc network
  cidr_blocks = ["192.168.0.0/16"]
}

resource "ucloud_subnet" "example" {
  name = "tf-example-subnet"
  tag  = "tf-example"

  # subnet's network must be contained by vpc network
  # and a subnet must have least 8 ip addresses in it (netmask < 30).
  cidr_block = "192.168.1.0/24"
  vpc_id     = ucloud_vpc.default.id
}
```

» Argument Reference

The following arguments are supported:

- `cidr_block` - (Required, ForceNew) The cidr block of the desired subnet, format in "0.0.0.0/0", such as: `192.168.0.0/24`.
- `vpc_id` - (Required, ForceNew) The id of the VPC that the desired subnet belongs to.

- `name` - (Optional) The name of the desired subnet. If not specified, terraform will auto-generate a name beginning with `tf-subnet`.
- `remark` - (Optional, ForceNew) The remarks of the subnet. (Default: `""`).
- `tag` - (Optional) A tag assigned to subnet, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and '.'. If

it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).

» Attributes Reference

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

- `create_time` - The time of creation of subnet, formatted in RFC3339 time string.

» Import

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

```
$ terraform import ucloud_subnet.example subnet-abc123456
```

» ucloud_nat_gateway

Provides a Nat Gateway resource.

» Example Usage

```
resource "ucloud_vpc" "foo" {
  name        = "tf-acc-nat-gateway-basic"
  tag         = "tf-acc"
  cidr_blocks = ["192.168.0.0/16"]
}

resource "ucloud_subnet" "foo" {
  name       = "tf-acc-nat-gateway-basic"
  tag        = "tf-acc"
  cidr_block = "192.168.1.0/24"
  vpc_id     = "${ucloud_vpc.foo.id}"
}

resource "ucloud_eip" "foo" {
  name          = "tf-acc-nat-gateway-basic"
  bandwidth     = 1
  internet_type = "bgp"
  charge_mode   = "bandwidth"
  tag           = "tf-acc"
}
```

```
data "ucloud_security_groups" "foo" {
  type = "recommend_web"
}

resource "ucloud_nat_gateway" "foo" {
  vpc_id            = ucloud_vpc.foo.id
  subnet_ids        = [ucloud_subnet.foo.id]
  eip_id            = ucloud_eip.foo.id
  name              = "tf-acc-nat-gateway-basic"
  tag               = "tf-acc"
  security_group    = data.ucloud_security_groups.foo.security_groups.0.id
}
```

» Argument Reference

The following arguments are supported:

- `vpc_id` - (Required, ForceNew) The ID of VPC linked to the Nat Gateway.
- `subnet_ids` - (Required) The list of subnet ID under the VPC.
- `eip_id` - (Required, ForceNew) The ID of eip associate to the Nat Gateway.
- `security_group` - (Required) The ID of the associated security group.
- `enable_white_list` - (Required) The boolean value to Controls whether or not start the whitelist mode.

- `white_list` - (Optional) The white list of instance under the Nat Gateway.
- `name` - (Optional, ForceNew) The name of the Nat Gateway which contains 6-63 characters and only support Chinese, English, numbers, '-', '_' and ''. If not specified, terraform will auto-generate a name beginning with `tf-nat-gateway-`.
- `remark` - (Optional, ForceNew) The remarks of the Nat Gateway. (Default: `""`).
- `tag` - (Optional, ForceNew) A tag assigned to Nat Gateway, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and ''. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).
- " ## Attributes Reference

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

- `create_time` - The time of creation of Nat Gateway, formatted in RFC3339 time string.

» Import

Nat Gateway can be imported using the `id`, e.g.

```
$ terraform import ucloud_nat_gateway.example natgw-abc123456
```

» ucloud_nat_gateway_rule

Provides a Nat Gateway resource.

» Example Usage

```
resource "ucloud_vpc" "foo" {
  name        = "tf-acc-nat-gateway-rule-basic"
  tag         = "tf-acc"
  cidr_blocks = ["192.168.0.0/16"]
}
resource "ucloud_subnet" "foo" {
  name       = "tf-acc-nat-gateway-rule-basic"
  tag        = "tf-acc"
  cidr_block = "192.168.1.0/24"
  vpc_id     = "${ucloud_vpc.foo.id}"
}

resource "ucloud_eip" "foo" {
  name          = "tf-acc-nat-gateway-rule-basic"
  bandwidth     = 1
  internet_type = "bgp"
  charge_mode   = "bandwidth"
  tag           = "tf-acc"
}

data "ucloud_security_groups" "foo" {
  type = "recommend_web"
}

data "ucloud_zones" "default" {}

data "ucloud_images" "default" {
  availability_zone = "${data.ucloud_zones.default.zones.0.id}"
  name_regex        = "^CentOS 7.[1-2] 64"
  image_type        = "base"
}
```

```
resource "ucloud_instance" "foo" {
  vpc_id            = ucloud_vpc.foo.id
  subnet_id         = ucloud_subnet.foo.id
  availability_zone = "${data.ucloud_zones.default.zones.0.id}"
  image_id          = "${data.ucloud_images.default.images.0.id}"
  instance_type     = "n-basic-1"
  charge_type       = "dynamic"
  name              = "tf-acc-nat-gateway-rule-basic"
  tag               = "tf-acc"
}

resource "ucloud_nat_gateway" "foo" {
  vpc_id            = ucloud_vpc.foo.id
  subnet_ids        = [ucloud_subnet.foo.id]
  eip_id            = ucloud_eip.foo.id
  name              = "tf-acc-nat-gateway-rule-basic"
  tag               = "tf-acc"
  enable_white_list = false
  security_group    = data.ucloud_security_groups.foo.security_groups.0.id
}

resource "ucloud_nat_gateway_rule" "foo" {
  nat_gateway_id = ucloud_nat_gateway.foo.id
  protocol       = "tcp"
  src_eip_id     = ucloud_eip.foo.id
  src_port_range = "88"
  dst_ip         = ucloud_instance.foo.private_ip
  dst_port_range = "80"
  name           = "tf-acc-nat-gateway-rule-basic"
}

resource "ucloud_nat_gateway_rule" "bar" {
  nat_gateway_id = ucloud_nat_gateway.foo.id
  protocol       = "tcp"
  src_eip_id     = ucloud_eip.foo.id
  src_port_range = "90-100"
  dst_ip         = ucloud_instance.foo.private_ip
  dst_port_range = "90-100"
  name           = "tf-acc-nat-gateway-rule-basic"
}
```

» Argument Reference

The following arguments are supported:

- `nat_gateway_id` - (Required, ForceNew) The ID of the Nat Gateway.
- `protocol` - (Required) The protocol of the Nat Gateway Rule. Possible values: `tcp`, `udp`.
- `src_eip_id` - (Required) The ID of eip associate to the Nat Gateway.
- `src_port_range` - (Required) The range of port numbers of the eip, range: 1-65535. (eg: `port` or `port1-port2`).
- `dst_ip` - (Required) The private ip of instance bound to the jNAT gateway.
- `dst_port_range` - (Required) The range of port numbers of the private ip, range: 1-65535. (eg: `port` or `port1-port2`).

- `name` - (Optional) The name of the Nat Gateway Rule which contains 6-63 characters and only support Chinese, English, numbers, '-', '_' and ''. If not specified, terraform will auto-generate a name beginning with `tf-nat-gateway-rule-`.

» ucloud_vip

Provides a VIP resource.

» Example Usage

```
resource "ucloud_vpc" "foo" {
    name        = "tf-acc-vip"
    tag         = "tf-acc"
    cidr_blocks = ["192.168.0.0/16"]
}
resource "ucloud_subnet" "foo" {
    name       = "tf-acc-vip"
    tag        = "tf-acc"
    cidr_block = "192.168.1.0/24"
    vpc_id     = ucloud_vpc.foo.id
}
resource "ucloud_vip" "foo" {
    vpc_id      = ucloud_vpc.foo.id
    subnet_id   = ucloud_subnet.foo.id
    name        = "tf-acc-vip-basic"
    remark      = "test"
}
```

» Argument Reference

The following arguments are supported:

- `vpc_id` - (Required, ForceNew) The ID of VPC linked to the VIP.
- `subnet_id` - (Required, ForceNew) The ID of subnet. If defined `vpc_id`, the `subnet_id` is Required.

- `name` - (Optional) The name of VIP. If not specified, terraform will auto-generate a name beginning with `tf-vip-`.
- `tag` - (Optional) A tag assigned to VIP, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and '.'. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).
- `remark` - (Optional) The remarks of the VIP. (Default: `""`).

» Attributes Reference

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

- `ip_address` - The ip address of the VIP.
- `create_time` - The time of creation for VIP, formatted in RFC3339 time string.

» ucloud_db_instances

This data source provides a list of database instance resources according to their database instance ID and name.

» Example Usage

```
data "ucloud_db_instances" "example" {}

output "first" {
  value = data.ucloud_db_instances.example.db_instances[0].id
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Optional) Availability zone where database instances are located. Such as: "cn-bj2-02". You may refer to list of availability zone
- `ids` - (Optional) A list of database instance IDs, all the database instances belong to this region will be retrieved if the ID is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting database instances by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).

» Attributes Reference

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

- `db_instances` - It is a nested type which documented below.
- `total_count` - Total number of database instances that satisfy the condition.

The attribute (`db_instances`) support the following:

- `availability_zone` - Availability zone where database instance is located.
- `id` - The ID of database instance.
- `name` - The name of database instance.
- `instance_type` - Specifies the type of database instance.
- `standby_zone` - Availability zone where the standby database instance is located for the high availability database instance with multiple zone.
- `vpc_id` - The ID of VPC linked to the database instances.
- `subnet_id` - The ID of subnet linked to the database instances.
- `engine` - The type of database instance engine.
- `engine_version` - The database instance engine version.
- `port` - The port on which the database instance accepts connections.
- `private_ip` - The private IP address assigned to the database instance.
- `instance_storage` - Specifies the allocated storage size in gigabytes (GB).
- `charge_type` - The charge type of db instance.
- `backup_count` - Specifies the number of backup saved per week.
- `backup_begin_time` - Specifies when the backup starts, measured in hour.
- `backup_date` - Specifies whether the backup took place from Sunday to Saturday by displaying 7 digits. 0 stands for backup disabled and 1 stands for backup enabled. The rightmost digit specifies whether the backup took place on Sunday, and the digits from right to left specify whether the backup took place from Monday to Saturday, it's mandatory required to backup twice per week at least. such as: digits "1100000" stands for the backup took place on Saturday and Friday.
- `backup_black_list` - The backup for database instance such as "test.%" or table such as "city.address" specified in the black lists are not supported.

- `tag` - A tag assigned to database instance.
- `status` - Specifies the status of database instance , possible values are: `Init`, `Fail`, `Starting`, `Running`, `Shutdown`, `Shutoff`, `Delete`, `Upgrading`, `Promoting`, `Recovering` and `Recover fail`.
- `create_time` - The creation time of database instance , formatted by RFC3339 time string.
- `expire_time` - The expiration time of database instance , formatted by RFC3339 time string.
- `modify_time` - The modification time of database instance , formatted by RFC3339 time string.

» ucloud_db_instance

Provides a Database instance resource.

Note Please do confirm if any task pending submission before reset your password, since the password reset will take effect immediately.

» Example Usagek

```
# Query availability zone
data "ucloud_zones" "default" {
}

# Create database instance
resource "ucloud_db_instance" "master" {
  name              = "tf-example-db"
  instance_storage  = 20
  instance_type     = "mysql-ha-1"
  engine            = "mysql"
  engine_version    = "5.7"
  password          = "2018_dbInstance"
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Required, ForceNew) Availability zone where database instance is located. Such as: ”cn-bj2-02”. You may refer to list of availability zone
- `engine` - (Required, ForceNew) The type of database engine, possible values are: ”mysql”, ”percona”.

- `engine_version` - (Required, ForceNew) The database engine version, possible values are: "5.5", "5.6", "5.7".
  - 5.5/5.6/5.7 for mysql and percona engine.
- `name` - (Optional) The name of database instance, which contains 6-63 characters and only support Chinese, English, numbers, '-', '_', '.', ',', '[', ']', ':'. If not specified, terraform will auto-generate a name beginning with `tf-db-instance`.
- `instance_storage` - (Required) Specifies the allocated storage size in gigabytes (GB), range from 20 to 4500GB. The volume adjustment must be a multiple of 10 GB. The maximum disk volume for SSD type are
  - 500GB if the memory chosen is equal or less than 6GB;
  - 1000GB if the memory chosen is from 8 to 16GB;
  - 2000GB if the memory chosen is 24GB or 32GB;
  - 3500GB if the memory chosen is 48GB or 64GB;
  - 4500GB if the memory chosen is equal or more than 96GB;
- `instance_type` - (Required) The type of database instance, please visit the instance type table.

- `standby_zone` - (Optional, ForceNew) Availability zone where the standby database instance is located for the high availability database instance with multiple zone; The disaster recovery of data center can be activated by switching to the standby database instance for the high availability database instance.
- `password` - (Optional) The password for the database instance which should have 8-30 characters. It must contain at least 3 items of Capital letters, small letter, numbers and special characters. The special characters include `-_`. If not specified, terraform will auto-generate a password.
- `port` - (Optional) The port on which the database accepts connections, the default port is 3306 for mysql and percona.
- `charge_type` - (Optional, ForceNew) The charge type of db instance, possible values are: `year`, `month` and `dynamic` as pay by hour (specific permission required). (Default: `month`).
- `duration` - (Optional, ForceNew) The duration that you will buy the db instance (Default: `1`). The value is `0` when pay by month and the instance will be vaild till the last day of that month. It is not required when `dynamic` (pay by hour).
- `vpc_id` - (Optional, ForceNew) The ID of VPC linked to the database instances.
- `subnet_id` - (Optional, ForceNew) The ID of subnet.
- `backup_count` - (Optional, ForceNew) Specifies the number of backup saved per week, it is 7 backups saved per week by default.
- `backup_begin_time` - (Optional) Specifies when the backup starts, measured in hour, it starts at one o'clock of 1, 2, 3, 4 in the morning by default.
- `backup_date` - (Optional) Specifies whether the backup took place from

Sunday to Saturday by displaying 7 digits. 0 stands for backup disabled and 1 stands for backup enabled. The rightmost digit specifies whether the backup took place on Sunday, and the digits from right to left specify whether the backup took place from Monday to Saturday, it's mandatory required to backup twice per week at least. such as: digits "1100000" stands for the backup took place on Saturday and Friday.

- `backup_black_list` - (Optional) The backup for database such as "test.%" or table such as "city.address" specified in the black lists are not supported.
- `tag` - (Optional, ForceNew) A tag assigned to database instance, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and '.'. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).

» Attributes Reference

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

- `status` - Specifies the status of database, possible values are: `Init`, `Fail`, `Starting`, `Running`, `Shutdown`, `Shutoff`, `Delete`, `Upgrading`, `Promoting`, `Recovering` and `Recover fail`.
- `private_ip` - The private IP address assigned to the database instance.
- `create_time` - The creation time of database, formatted by RFC3339 time string.
- `expire_time` - The expiration time of database, formatted by RFC3339 time string.
- `modify_time` - The modification time of database, formatted by RFC3339 time string.

» Import

DB Instance can be imported using the `id`, e.g.

```
$ terraform import ucloud_db_instance.example udbha-abc123456
```

» ucloud_redis_instance

The UCloud Redis instance is a key-value online storage service compatible with the Redis protocol.

» Example Usage

```
data "ucloud_zones" "default" {}
```

```
resource "ucloud_redis_instance" "master" {
  availability_zone = data.ucloud_zones.default.zones[0].id
  instance_type     = "redis-master-2"
  password          = "2018_Tfacc"
  engine_version    = "4.0"

  name = "tf-example-redis-master"
  tag  = "tf-example"
}

resource "ucloud_redis_instance" "distributed" {
  availability_zone = data.ucloud_zones.default.zones[0].id
  instance_type     = "redis-distributed-16"

  name = "tf-example-redis-distributed"
  tag  = "tf-example"
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Required, ForceNew) Availability zone where Redis instance is located. Such as: "cn-bj2-02". You may refer to list of availability zone
- `instance_type` - (Required) The type of Redis instance, please visit the instance type table for more details.

- `name` - (Optional) The name of Redis instance, which contains 6-63 characters and only support English, numbers, '-', '_'. If not specified, terraform will auto-generate a name beginning with `tf-redis-instance`.
- `charge_type` - (Optional, ForceNew) The charge type of Redis instance, possible values are: `year`, `month` and `dynamic` as pay by hour (specific permission required). (Default: `month`).
- `duration` - (Optional, ForceNew) The duration that you will buy the Redis instance (Default: `1`). The value is `0` when pay by month and the instance will be valid till the last day of that month. It is not required when `dynamic` (pay by hour).
- `tag` - (Optional, ForceNew) A tag assigned to Redis instance, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and '.'. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).

- `vpc_id` - (Optional, ForceNew) The ID of VPC linked to the Redis instance.
- `subnet_id` - (Optional, ForceNew) The ID of subnet linked to the Redis instance.
- `engine_version` - (active-standby Redis Required, ForceNew) The version of engine of active-standby Redis. Possible values are: 3.0, 3.2, 4.0 and 5.0.
- `password` - (Optional) The password for active-standby Redis instance which should have 6-36 characters. It must contain at least 3 items of Capital letters, small letter, numbers and special characters. The special characters include `-_`.

Note The active-standby Redis doesn't support to be created on multiple zones with Terraform.

» Attributes Reference

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

- `ip_set` - ip_set is a nested type. ip_set documented below.
- `create_time` - The creation time of Redis instance, formatted by RFC3339 time string.
- `expire_time` - The expiration time of Redis instance, formatted by RFC3339 time string.
- `status` - The status of KV Redis instance.

The attribute (`ip_set`) support the following:

- `ip` - The virtual ip of Redis instance.
- `port` - The port on which Redis instance accepts connections, it is 6379 by default.

» ucloud_memcache_instance

The UCloud Memcache instance is a key-value online storage service compatible with the Memcached protocol.

» Example Usage

```
data "ucloud_zones" "default" {}

resource "ucloud_memcache_instance" "master" {
  availability_zone = data.ucloud_zones.default.zones[0].id
```

```
  instance_type     = "memcache-master-2"

  name = "tf-example-memcache"
  tag  = "tf-example"
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Required, ForceNew) Availability zone where Memcache instance is located. Such as: "cn-bj2-02". You may refer to list of availability zone
- `instance_type` - (Required) The type of Memcache instance, please visit the instance type table for more details.

- `name` - (Optional) The name of Memcache instance, which contains 6-63 characters and only support English, numbers, '-', '_'. If not specified, terraform will auto-generate a name beginning with `tf-memcache-instance`.
- `charge_type` - (Optional, ForceNew) The charge type of Memcache instance, possible values are: `year`, `month` and `dynamic` as pay by hour (specific permission required). (Default: `month`).
- `duration` - (Optional, ForceNew) The duration that you will buy the Memcache instance (Default: `1`). The value is `0` when pay by month and the instance will be valid till the last day of that month. It is not required when `dynamic` (pay by hour).
- `tag` - (Optional, ForceNew) A tag assigned to Memcache instance, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and ''. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).
- `vpc_id` - (Optional, ForceNew) The ID of VPC linked to the Memcache instance.
- `subnet_id` - (Optional, ForceNew) The ID of subnet linked to the Memcache instance.

» Attributes Reference

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

- `ip_set` - ip_set is a nested type. ip_set documented below.
- `create_time` - The creation time of Memcache instance, formatted by RFC3339 time string.
- `expire_time` - The expiration time of Memcache instance, formatted by RFC3339 time string.

- `status` - The status of KV Memcache instance.

The attribute (`ip_set`) support the following:

- `ip` - The virtual ip of Memcache instance.
- `port` - The port on which Memcache instance accepts connections, it is 6379 by default.

» ucloud_vpn_gateways

This data source providers a list of VPN Gateway resources according to their ID, name, vpc and tag.

» Example Usage

```
data "ucloud_vpn_gateways" "example" {
}

output "first" {
  value = data.ucloud_vpn_gateways.example.vpn_gateways[0].id
}
```

» Argument Reference

The following arguments are supported:

- `ids` - (Optional) A list of VPN Gateway IDs, all the VPN Gateways belongs to the defined region will be retrieved if this argument is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting VPN Gateways by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).
- `tag` - (Optional) A tag assigned to VPN Gateway.
- `vpc_id` - (Optional) The ID of VPC linked to the VPN Gateway.

» Attributes Reference

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

- `vpn_gateways` - It is a nested type. VPN Gateways documented below.
- `total_count` - Total number of VPN Gateways that satisfy the condition.

The attribute (`vpn_gateways`) support the following:

- `id` - The ID of VPN Gateway.
- `name` - The name of the VPN Gateway.
- `remark` - The remarks of VPN Gateway.
- `tag` - A tag assigned to the VPN Gateway.
- `grade` - The type of the VPN Gateway.
- `vpc_id` - The ID of VPC linked to the VPN Gateway.
- `charge_type` - The charge type of VPN Gateway.
- `auto_renew` - Whether to renew an VPN Gateway automatically or not.
- `create_time` - The time of creation for VPN Gateway, formatted in RFC3339 time string.
- `expire_time` - The expiration time for VPN Gateway, formatted in RFC3339 time string.
- `ip_set` - It is a nested type which documented below.

The attribute (`ip_set`) supports the following:

- `internet_type` - Type of Elastic IP routes.
- `ip` - Elastic IP address.

» ucloud_vpn_customer_gateways

This data source providers a list of VPN Customer Gateway resources according to their ID, name and tag.

» Example Usage

```
data "ucloud_vpn_customer_gateways" "example" {
}

output "first" {
  value = data.ucloud_vpn_customer_gateways.example.vpn_customer_gateways[0].id
}
```

» Argument Reference

The following arguments are supported:

- `ids` - (Optional) A list of VPN Customer Gateway IDs, all the VPN Customer Gateways belongs to the defined region will be retrieved if this argument is `[]`.

- `name_regex` - (Optional) A regex string to filter resulting VPN Customer Gateways by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).
- `tag` - (Optional) A tag assigned to VPN Customer Gateway.

» Attributes Reference

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

- `vpn_customer_gateways` - It is a nested type. VPN Customer Gateways documented below.
- `total_count` - Total number of VPN Customer Gateways that satisfy the condition.

The attribute (`vpn_customer_gateways`) support the following:

- `id` - The ID of VPN Customer Gateway.
- `name` - The name of the VPN Customer Gateway.
- `remark` - The remarks of VPN Customer Gateway.
- `tag` - A tag assigned to the VPN Customer Gateway.
- `ip_address` - The ip address of the VPN Customer Gateway.
- `create_time` - The time of creation for VPN Customer Gateway, formatted in RFC3339 time string.

» ucloud_vpn_connections

This data source providers a list of VPN Connection resources according to their ID, name and tag.

» Example Usage

```
data "ucloud_vpn_connections" "example" {
}

output "first" {
  value = data.ucloud_vpn_connections.example.vpn_connections[0].id
}
```

» Argument Reference

The following arguments are supported:

- `ids` - (Optional) A list of VPN Connection IDs, all the VPN Connections belongs to the defined region will be retrieved if this argument is `[]`.
- `name_regex` - (Optional) A regex string to filter resulting VPN Connections by name.
- `output_file` - (Optional) File name where to save data source results (after running `terraform plan`).
- `tag` - (Optional) A tag assigned to VPN Connection.
- `vpc_id` - (Optional) The ID of VPC linked to the VPN Connection.

» Attributes Reference

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

- `vpn_connections` - It is a nested type. VPN Connections documented below.
- `total_count` - Total number of VPN Connections that satisfy the condition.

The attribute (`vpn_connections`) support the following:

- `id` - The ID of VPN Connection.
- `name` - The name of the VPN Connection.
- `remark` - The remarks of VPN Connection.
- `tag` - A tag assigned to the VPN Connection.
- `vpn_gateway_id` - The ID of VPN Gateway.
- `customer_gateway_id` - The ID of VPN Customer Gateway.
- `vpc_id` - The ID of VPC linked to the VPN Connection.
- `create_time` - The time of creation for VPN Connection, formatted in RFC3339 time string.
- `ike_config` - It is a nested type which documented below.
- `ipsec_config` - It is a nested type which documented below.

The attribute (`ike_config`) supports the following:

- `pre_shared_key` - The key used for authentication between the VPN gateway and the Customer gateway.
- `ike_version` - The version of the IKE protocol.
- `exchange_mode` - The negotiation exchange mode of IKE V1 of VPN gateway.
- `encryption_algorithm` - The encryption algorithm of IKE negotiation.
- `authentication_algorithm` - The authentication algorithm of IKE negotiation.
- `local_id` - The identification of the VPN gateway.
- `remote_id` - The identification of the Customer gateway.
- `dh_group` - The Diffie-Hellman group used by IKE negotiation.

- `sa_life_time` - The Security Association lifecycle as the result of IKE negotiation.

The attribute (`ipsec_config`) supports the following:

- `local_subnet_ids` - The id list of Local subnet.
- `remote_subnets` - The ip address list of remote subnet.
- `protocol` - The security protocol of IPSec negotiation.
- `encryption_algorithm` - The encryption algorithm of IPSec negotiation.
- `authentication_algorithm` - The authentication algorithm of IPSec negotiation.
- `pfs_dh_group` - Whether the PFS of IPSec negotiation is on or off, `disable` as off, The Diffie-Hellman group as open.
- `sa_life_time` - The Security Association lifecycle as the result of IPSec negotiation.
- `sa_life_time_bytes` - The Security Association lifecycle in bytes as the result of IPSec negotiation.

» ucloud_vpn_gateway

Provides a VPN Gateway resource.

» Example Usage

```
resource "ucloud_vpc" "foo" {
    name        = "tf-acc-vpn-gateway-basic"
    tag         = "tf-acc"
    cidr_blocks = ["192.168.0.0/16"]
}

resource "ucloud_eip" "foo" {
    name          = "tf-acc-vpn-gateway-basic"
    bandwidth     = 1
    internet_type = "bgp"
    charge_mode   = "bandwidth"
    tag           = "tf-acc"
}

resource "ucloud_vpn_gateway" "foo" {
    vpc_id      = ucloud_vpc.foo.id
    grade       = "enhanced"
    eip_id      = ucloud_eip.foo.id
    name        = "tf-acc-vpn-gateway-basic"
    tag         = "tf-acc"
```

```
}
```

» Argument Reference

The following arguments are supported:

- `vpc_id` - (Required, ForceNew) The ID of VPC linked to the VPN Gateway.
- `grade` - (Required) The type of the VPN Gateway. Possible values: `standard`, `enhanced`. `standard` recommended application scenario: Applicable to services with bidirectional peak bandwidth of 1M~50M; `enhanced` recommended application scenario: Suitable for services with bidirectional peak bandwidths of 50M~100M.
- `eip_id` - (Required, ForceNew) The ID of eip associate to the VPN Gateway.
- `security_group` - (Required) The ID of the associated security group.

- `charge_type` - (Optional, ForceNew) The charge type of VPN Gateway, possible values are: `year`, `month` and `dynamic` as pay by hour (specific permission required). (Default: `month`).
- `duration` - (Optional, ForceNew) The duration that you will buy the VPN Gateway (Default: `1`). The value is `0` when pay by month and the instance will be valid till the last day of that month. It is not required when `dynamic` (pay by hour).
- `name` - (Optional, ForceNew) The name of the VPN Gateway which contains 1-63 characters and only support Chinese, English, numbers, '-', '_' and ''. If not specified, terraform will auto-generate a name beginning with `tf-vpn-gateway-`.
- `remark` - (Optional, ForceNew) The remarks of the VPN Gateway. (Default: `""`).
- `tag` - (Optional, ForceNew) A tag assigned to VPN Gateway, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and ''. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).
- " ## Attributes Reference

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

- `create_time` - The creation time for VPN Gateway, formatted in RFC3339 time string.
- `expire_time` - The expiration time for VPN Gateway, formatted in RFC3339 time string.

» Import

VPN Gateway can be imported using the `id`, e.g.

```
$ terraform import ucloud_vpn_gateway.example vpngw-abc123456
```

» ucloud_vpn_customer_gateway

Provides a VPN Customer Gateway resource.

» Example Usage

```
resource "ucloud_vpn_customer_gateway" "foo" {
    ip_address  = "10.0.0.1"
    name        = "tf-acc-vpn-customer-gateway-basic"
    tag         = "tf-acc"
}
```

» Argument Reference

The following arguments are supported:

- `ip_address` - (Required, ForceNew) The ip address of the VPN Customer Gateway.

- `name` - (Optional, ForceNew) The name of the VPN Customer Gateway which contains 1-63 characters and only support Chinese, English, numbers, '-', '_' and ''. If not specified, terraform will auto-generate a name beginning with `tf-vpn-customer-gateway-`.
- `remark` - (Optional, ForceNew) The remarks of the VPN Customer Gateway. (Default: `""`).
- `tag` - (Optional, ForceNew) A tag assigned to VPN Customer Gateway, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and ''. If it is not filled in or a empty string is filled in, then default tag will be assigned. (Default: `Default`).
- " ## Attributes Reference

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

- `create_time` - The creation time for VPN Customer Gateway, formatted in RFC3339 time string.

» Import

VPN Customer Gateway can be imported using the `id`, e.g.

```
$ terraform import ucloud_vpn_gateway.example remotevpngw-abc123456
```

» ucloud_vpn_connection

Provides a IPSec VPN Gateway Connection resource.

» Example Usage

```
resource "ucloud_vpc" "foo" {
  name        = "tf-acc-vpn-connection-basic"
  tag         = "tf-acc"
  cidr_blocks = ["192.168.0.0/16"]
}

resource "ucloud_subnet" "foo" {
  name       = "tf-acc-vpn-connection-basic"
  tag        = "tf-acc"
  cidr_block = "192.168.1.0/24"
  vpc_id     = "${ucloud_vpc.foo.id}"
}

resource "ucloud_eip" "foo" {
  name          = "tf-acc-vpn-connection-basic"
  bandwidth     = 1
  internet_type = "bgp"
  charge_mode   = "bandwidth"
  tag           = "tf-acc"
}

resource "ucloud_vpn_gateway" "foo" {
  vpc_id = ucloud_vpc.foo.id
  grade  = "standard"
  eip_id = ucloud_eip.foo.id
  name   = "tf-acc-vpn-connection-basic"
  tag    = "tf-acc"
}

resource "ucloud_vpn_customer_gateway" "foo" {
  ip_address = "10.0.0.1"
  name       = "tf-acc-vpn-connection-basic"
```

```
  tag        = "tf-acc"
}

resource "ucloud_vpn_connection" "foo" {
  vpn_gateway_id      = ucloud_vpn_gateway.foo.id
  customer_gateway_id = ucloud_vpn_customer_gateway.foo.id
  vpc_id              = ucloud_vpc.foo.id
  name                = "tf-acc-vpn-connection-basic"
  tag                 = "tf-acc"
  remark              = "test"
  ike_config {
    pre_shared_key = "test_2019"
  }

  ipsec_config {
    local_subnet_ids = [ucloud_subnet.foo.id]
    remote_subnets   = ["10.0.0.0/24"]
  }
}
```

» Argument Reference

The following arguments are supported:

- `vpc_id` - (Required, ForceNew) The ID of VPC linked to the VPN Gateway Connection.
- `vpn_gateway_id` - (Required, ForceNew) The ID of the VPN Customer Gateway.
- `customer_gateway_id` - (Required, ForceNew) The grade of the VPN Gateway
- `ike_config` - (Required) The configurations of IKE negotiation. Each ike_config supports fields documented below.
- `ipsec_config` - (Required) The configurations of IPSec negotiation. Each ipsec_config supports fields documented below.

- `name` - (Optional) The name of the VPN Gateway Connection which contains 1-63 characters and only support Chinese, English, numbers and special characters: `-_.`. If not specified, terraform will auto-generate a name beginning with `tf-vpn-connection-`.
- `remark` - (Optional) The remarks of the VPN Gateway Connection. (Default: `""`).
- `tag` - (Optional, ForceNew) A tag assigned to VPN Gateway Connection, which contains at most 63 characters and only support Chinese, English, numbers, '-', '_', and ''. If it is not filled in or a empty string is filled in,

then default tag will be assigned. (Default: `Default`).

» Block ike_config

The ike_config mapping supports the following:

- `pre_shared_key` - (Required) The key used for authentication between the VPN gateway and the Customer gateway which contains 1-128 characters and only support English, numbers and special characters: `!@#$%^&*()_+-=[]:,./'~`.
- `ike_version` - (Optional) The version of the IKE protocol which only be supported IKE V1 protocol at present. Possible values: ikev1. (Default: ikev1)
- `exchange_mode` - (Optional) The negotiation exchange mode of IKE V1 of VPN gateway. Possible values: `main` (main mode), `aggressive` (aggressive mode). (Default: `main`)
- `encryption_algorithm` - (Optional) The encryption algorithm of IKE negotiation. Possible values: `aes128`, `aes192`, `aes256`, `aes512`, `3des`. (Default: `aes128`).
- `authentication_algorithm` - (Optional) The authentication algorithm of IKE negotiation. Possible values: `sha1`, `md5`, `sha2-256`. (Default: `sha1`)
- `local_id` - (Optional) The identification of the VPN gateway.
- `remote_id` - (Optional) The identification of the Customer gateway.
- `dh_group` - (Optional) The Diffie-Hellman group used by IKE negotiation. Possible values: `1`, `2`, `5`, `14`, `15`, `16`. (Default:`15`)
- `sa_life_time` - (Optional) The Security Association lifecycle as the result of IKE negotiation. Unit: second. Range: 600-604800. (Default: `86400`)

» Block ipsec_config

The ipsec_config mapping supports the following:

- `local_subnet_ids` - (Required) The id list of Local subnet.
- `remote_subnets` - (Required) The ip address list of remote subnet.
- `protocol` - (Optional) The security protocol of IPSec negotiation. Possible values: `esp`, `ah`. (Default:`esp`)
- `encryption_algorithm` - (Optional) The encryption algorithm of IPSec negotiation. Possible values: `aes128`, `aes192`, `aes256`, `aes512`, `3des`. (Default: `aes128`).
- `authentication_algorithm` - (Optional) The authentication algorithm of IPSec negotiation. Possible values: `sha1`, `md5`. (Default: `sha1`)
- `pfs_dh_group` - (Optional) Whether the PFS of IPSec negotiation is on or off, `disable` as off, The Diffie-Hellman group as open. Possible values: `disable`, `1`, `2`, `5`, `14`, `15`, `16`. (Default:`disable`)

- `sa_life_time` - (Optional) The Security Association lifecycle as the result of IPSec negotiation. Unit: second. Range: 1200-604800. (Default: `3600`)
- `sa_life_time_bytes` - (Optional) The Security Association lifecycle in bytes as the result of IPSec negotiation. Unit: second. Range: 1200-604800. (Default: `3600`)

» Attributes Reference

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

- `create_time` - The creation time for VPN Gateway Connection, formatted in RFC3339 time string.
- `expire_time` - The expiration time for VPN Gateway Connection, formatted in RFC3339 time string.

» Import

VPN Connection can be imported using the `id`, e.g.

```
$ terraform import ucloud_vpn_connection.example vpntunnel-abc123456
```

» ucloud_udpn_connection

UDPN (UCloud Dedicated Private Network) you can use Dedicated Private Network to achieve high-speed, stable, secure, and dedicated communications between different data centers. The most frequent scenario is to create network connection of clusters across regions.

VPC Peering Connections with UDPN Connection The cross-region Dedicated Private Network must be established if the two VPCs located in different regions are expected to be connected.

Note The additional packet head will be added and included in the overall length of packet due to the tunneling UDPN adopted. Since the number of the bytes of packet head is fixed, the bigger data packet is, the less usage will be taken for the packet head.

» Example Usage

```
provider "ucloud" {
  region = "cn-bj2"
}
```

```
// connect provider's region (cn-bj2) and peer region (cn-sh2)
resource "ucloud_udpn_connection" "example" {
  bandwidth   = 2
  peer_region = "cn-sh2"
}
```

» Argument Reference

The following arguments are supported:

- `bandwidth` - (Optional) Maximum bandwidth to the elastic public network, measured in Mbps (Mega bit per second). range from 2 - 1000M. The default value is "1"
- `duration` - (Optional, ForceNew) The duration that you will buy the resource, the default value is "1". It is not required when "dynamic" (pay by hour), the value is "0" when pay by month and the instance will be valid till the last day of that month.
- `charge_type` - (Optional, ForceNew) Charge type. Possible values are: "year" as pay by year, "month" as pay by month, "dynamic" as pay by hour. The default value is "month".
- `peer_region` - (Optional, ForceNew) The correspondent region of dedicated connection, please refer to the region and availability zone list and UDPN price list.

» Attributes Reference

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

- `create_time` - The time of creation for UDPN connection, formatted by RFC3339 time string.
- `expire_time` - The expiration time for UDPN connection, formatted by RFC3339 time string.

» Import

UDPN connection can be imported using the `id`, e.g.

```
$ terraform import ucloud_udpn_connection.example udpn-abc123456
```