

Null Provider

The `null` provider is a rather-unusual provider that has constructs that intentionally do nothing. This may sound strange, and indeed these constructs do not need to be used in most cases, but they can be useful in various situations to help orchestrate tricky behavior or work around limitations.

The documentation of each feature of this provider, accessible via the navigation, gives examples of situations where these constructs may prove useful.

Usage of the `null` provider can make a Terraform configuration harder to understand. While it can be useful in certain cases, it should be applied with care and other solutions preferred when available.

Null Resource

The `null_data_source` data source implements the standard data source lifecycle but does not interact with any external APIs.

Example Usage

The primary use-case for the null data source is to gather together collections of intermediate values to re-use elsewhere in configuration:

```
data "null_data_source" "values" {
  inputs = {
    all_server_ids = "${concat(aws_instance.green.*.id, aws_instance.blue.*.id)}"
    all_server_ips = "${concat(aws_instance.green.*.private_ip, aws_instance.blue.*.private_ip)}"
  }
}

resource "aws_elb" "main" {
  # ...

  instances = "${data.null_data_source.values.outputs["all_server_ids"]}"
}

output "all_server_ids" {
  value = "${data.null_data_source.values.outputs["all_server_ids"]}"
}

output "all_server_ips" {
  value = "${data.null_data_source.values.outputs["all_server_ips"]}"
}

# ... (other uses of the values) ...
```

Argument Reference

The following arguments are supported:

- `inputs` - (Optional) A map of arbitrary strings that is copied into the `outputs` attribute, and accessible directly for interpolation.
- `has_computed_default` - (Optional) If set, its literal value will be stored and returned. If not, its value defaults to `"default"`. This argument exists primarily for testing and has little practical use.

Attributes Reference

The following attributes are exported:

- `outputs` - After the data source is "read", a copy of the `inputs` map.

- `random` - A random value. This is primarily for testing and has little practical use; prefer the `random` provider (</docs/providers/random/>) for more practical random number use-cases.

Null Resource

The `null_resource` resource implements the standard resource lifecycle but takes no further action.

The `triggers` argument allows specifying an arbitrary set of values that, when changed, will cause the resource to be replaced.

Example Usage

The primary use-case for the null resource is as a do-nothing container for arbitrary actions taken by a provisioner, as follows:

```
resource "aws_instance" "cluster" {
  count = 3

  # ...
}

resource "null_resource" "cluster" {
  # Changes to any instance of the cluster requires re-provisioning
  triggers = {
    cluster_instance_ids = "${join(",", aws_instance.cluster.*.id)}"
  }

  # Bootstrap script can run on any instance of the cluster
  # So we just choose the first in this case
  connection {
    host = "${element(aws_instance.cluster.*.public_ip, 0)}"
  }

  provisioner "local-exec" {
    # Bootstrap script called with private_ip of each node in the cluster
    command = "bootstrap-cluster.sh ${join(" ", aws_instance.cluster.*.private_ip)}"
  }
}
```

In this example, three EC2 instances are created and then a `null_resource` instance is used to gather data about all three and execute a single action that affects them all. Due to the `triggers` map, the `null_resource` will be replaced each time the instance ids change, and thus the `remote-exec` provisioner will be re-run.

Argument Reference

The following arguments are supported:

- `triggers` - (Optional) A map of arbitrary strings that, when changed, will force the null resource to be replaced, re-running any associated provisioners.

Attributes Reference

The following attributes are exported:

- `id` - An arbitrary value that changes each time the resource is replaced. Can be used to cause other resources to be updated or replaced in response to `null_resource` changes.