How are data sources used in Terraform?

Asked 5 years, 1 month ago Modified 1 year, 8 months ago Viewed 86k times Part of AWS Collective

84

The Terraform Data Sources documentation tells me what a data source is, but I do not quite understand it. Can somebody give me a use case of data source? What is the difference between it and configuring something using variables?

amazon-web-services terraform

Share Improve this question

Follow

edited Jan 23, 2019 at 15:12

d4nyll

11.4k 6 53 66

asked Dec 8, 2017 at 20:20

user389955

9,043 10 50 88

4 good question. I also hadn't understood it until I read it for a 3rd time in the 3rd day... – Jing He Jul 21, 2021 at 16:46

4 Answers

Sorted by:

Highest score (default)

90

Data sources can be used for a number of reasons; but their goal is to **do something** and then **give you data**.

Let's take the example from their documentation:

```
# Find the latest available AMI that is tagged with Component = web
data "aws_ami" "web" {
  filter {
    name   = "state"
    values = ["available"]
  }

  filter {
    name   = "tag:Component"
    values = ["web"]
  }

  most_recent = true
}
```

This uses the aws_ami data source - this is different than a resource! It will instead just give you information, and not create anything. This example in particular will call out to the `describe-images` AWS API call, pass in a few `--filter` options as specified, and return an object that you can get information from - take a look at these attributes!

- owner_id
- description
- image_id

... The list goes on. This is really useful if I were, let's say - always wanting to pull the latest AMI matching some tags, and keep a launch configuration up to date with it. I could use this data provider rather than always have to update a variable or hard-code the ID.

Data source can be used for other reasons as well; one of my favorites is the template provider.

Good luck!

Share Improve this answer Follow

answered Dec 8, 2017 at 20:28

TJ Biddle
5,738 4 39 47

2 HI, Thanks TJ. your answer is very clear. I do not understand why terraform does not explain what is data source in your way. :-) – user389955 Dec 8, 2017 at 21:54

1 more question which may or may not related to data source: I have created a few new services (using terraform) in an existing AWS cloud (manually created). Right now, I indicated the ids of existing cloud (subnet, role, vpc id, etc) in my service resources, e.g. subnet_id = "${var.subnet_private1_id}", and I set subnet_private1_id as the actual subnet id in terraform.tfvars In the future I will run my terraform code to another AWS cloud. then all id I related to the current cloud will be re-configured. Should I just set all related id of current cloud as variables, or shall I use datasource? – user389955 Dec 8, 2017 at 22:01

You want to use `terraform import` – TJ Biddle Dec 8, 2017 at 23:52

T J Biddle: I do not need to change the existing things, I only need to create new stuffs based on the existing cloud so I do not need to import. see stackoverflow.com/questions/47665428/… – user389955 Dec 9, 2017 at 20:38

Gotchya - Well I really recommend trying to get as much of your infrastructure into code as possible; it will make things a lot easier down the road. – TJ Biddle Dec 9, 2017 at 21:11

49

Data sources provide information about entities that are *not managed by the current Terraform configuration*.

This may include:

- Configuration data from Consul
- Information about the state of manually-configured infrastructure components

In other words, data sources are *read-only* views into the state of pre-existing components external to our configuration.

For example, let's suppose we want to create a Terraform configuration for a new AWS EC2 instance. We want to use an AMI image which were created and uploaded by a Jenkins job using the AWS CLI, and not managed by Terraform. As part of the configuration for our Jenkins job, this AMI image will always have a name with the prefix `app-` .

In this case, we can use the `aws_ami` data source to obtain information about the most recent AMI image that has the name prefix `app-` .

```
data "aws_ami" "app_ami" {
  most_recent = true
  filter {
    name   = "name"
    values = ["app-*"]
  }
}
```

Data sources export attributes, just like resources do. We can interpolate these attributes using the syntax `data.TYPE.NAME.ATTR` . In our example, we can interpolate the value of the AMI ID as `data.aws_ami.app_ami.id` , and pass it as the `ami` argument for our `aws_instance` resource.

```
resource "aws_instance" "app" {
  ami           = "${data.aws_ami.app_ami.id}"
  instance_type = "t2.micro"
}
```

Data sources are most powerful when retrieving information about *dynamic* entities - those whose properties change value often. For example, the next time Terraform fetches data for our `aws_ami` data source, the value of the exported attributes may be different (we might have built and pushed a new AMI).

Variables are used for *static* values, those that rarely changes, such as your access and secret keys, or a standard list of sudoers for your servers.

Share Improve this answer Follow

answered Jan 23, 2019 at 15:24

d4nyll
11.4k 6 53 66

Thanks d4nyll. In another word, 1) data source is to provide information of existing infrastructure not to create new services. 2) and it retrieves dynamic entities based on your filtering logic. – user389955 Jan 26, 2019 at 2:18

1 1) Yes! 2) Each data source is different. For the `aws_ami` data source, because it actually called the `describe-images` CLI command in the back, and *it* supports filtering. For other data sources, there may not be filtering. The configuration of each data source is specific to that data source and provider. – d4nyll Jan 26, 2019 at 13:05

Oh. Good to know. Thanks again @d4nyll – user389955 Jan 28, 2019 at 17:20

11, 2019 at 15:24

@Averell that's a data source that I haven't used before, and it does seem like it goes against the norm. If you find any other examples, let me know, I will update my answer. I guess a more accurate way of describing data sources is that they get the state of existing resources and incorporate it as part of the Terraform state. The `archive_file` 's existing resource is a file or directory of files, and it seems like the entire file content are stored in the state. – d4nyll Oct 11, 2019 at 21:31

15

Good examples up there!

The main difference between Terraform data source, resource and variable is :

Resource: Provisioning of resources/infra on our platform. Create, Update and delete!

Variable Provides predefined values as variables on our IAC. Used by resource for provisioning.

Data Source: Fetch values from our infra/provider and and provides data for our resource to provision infra/resource.

Examples are well explained above :)

Share Improve this answer Follow

edited May 20, 2021 at 12:46

answered May 20, 2021 at 12:37

Kailash
179 1 4

9

Data sources are used to fetch the data from the provider end, so that It can be used as configuration in .tf files, Instead of hardcoding it. Example: Below code fetches the AWS AMI ID and uses it to launch AWS instance.

```
data "aws_ami" "std_ami" {
  most_recent = true
  owners      = ["amazon"]

filter {
    name   = "root-device-type"
    values = ["ebs"]
  }

filter {
    name   = "virtualization-type"
    values = ["hvm"]
  }
}

resource "aws_instance" "myec2" {
  ami           = data.aws_ami.std_ami.id
  instance_type = "t2.micro"
}
```

Shiva Garg
756 8 16