



DevOps 2021:

- Terraform Basics and Essentials - Practical

Чем будем маяться :

- ***Terraform***
 - ***Root template - variables, .tfvars, outputs, backend***
 - ***Modules***
 - ***plan, apply, destroy, state, output, fmt***
 - ***Организация Terraform***
 - ***Best Practice***
 - ***Testing & Debug***



Terraform Basics and Essentials

Resource and Data Sources:

```
resource "<resource_type>" "<resource_name>" {  
  <resource_parameter1>=<value1>  
  <resource_parameter2>=<value2>  
}
```

```
data "<resource_type>" "<data_name>" {  
  <data_parameter1>=<value1>  
}
```

Reference in templates:

<resource_type>.<resource_name>.<attribute_name>

Eg.

aws_instance.demo.id



Terraform Basics and Essentials

Variables.tf and locals.tf:

```
variable "<variable_name>" {  
  description = <variable_description>  
  default     = "<default_value>" # if not set will be required to pass during apply  
  type       = string|bool|map  
}
```

```
locals {  
  <local_variable_name> = <local_variable_value>  
}
```

Reference in templates:

var.testvar # variable reference
local.testvar # local reference

.tfvars:

<var> = <value format>

Pass variables with tfvars file during apply

terraform apply -var-file=<tfvars_file_name>.tfvars



Terraform Basics and Essentials

Providers:

```
provider "aws" {  
  region = var.region  
  version = "~> 3.0"  
}  
  
terraform {  
  required_providers {  
    aws = {  
      source = "hashicorp/aws"  
      version = "~> 3.0"  
    }  
  }  
}
```

Backend.tf:

```
terraform {  
  backend "s3" {  
    bucket = "goldfire-non-prod-terraform-states"  
    key = "dev/ec2.tfstate"  
    region = "us-west-2"  
    dynamodb_table = "terraform-state-lock"  
  }  
}
```

Backend initialize first, you cannot use variables in backend

Functions:

- `tolist(["a", "b", "c"]) -> ["a", "b", "c",]`
- `format("Hello, %s!", var.name) -> Hello, foo!`
- `join(", ", ["foo", "bar", var.baz]) -> foo, bar, baz`
- `length(["a", "b"]) -> 2`

Best Practice :

- *Разделяй окружения и аккаунты по папкам*
- *Разделяй **state** по ресурсам (**EC2, RDS, Lambda etc**)*
- *Используй **remote_state***
- *Следи и обозначай зависимости в документации*
- *Выноси в **variables** или **locals** все значения*

```
glebchuev ~/Andersen/Examples/Terraform_Example_EC2 tree
.
├── dev
│   ├── ec2_demo
│   │   ├── backend.tf
│   │   ├── ec2_test.tf
│   │   ├── env.tf -> ../env.tf
│   │   ├── iam.tf
│   │   ├── key-pair.tf
│   │   ├── main.tf
│   │   ├── null_resource.tf
│   │   ├── outputs.tf
│   │   ├── remote_state.tf
│   │   ├── secret_vars.tf
│   │   ├── secrets.tfvars
│   │   ├── sg.tf
│   │   ├── templates
│   │   │   └── user_data.sh.tpl
│   │   ├── terraform.tfvars
│   │   └── variables.tf
│   ├── env.tf
│   └── vpc_demo
│       ├── backend.tf
│       ├── env.tf -> ../env.tf
│       ├── main.tf
│       ├── outputs.tf
│       ├── terraform.tfvars
│       └── variables.tf
├── modules
│   ├── ec2-with-recovery
│   │   ├── cloudwatch_metric_alarm.tf
│   │   ├── data_readers.tf
│   │   ├── eip.tf
│   │   ├── iam_instance_profile.tf
│   │   ├── locals.tf
│   │   ├── main.tf
│   │   ├── outputs.tf
│   │   ├── sg.tf
│   │   └── variables.tf
│   └── vpc
│       ├── main.tf
│       └── outputs.tf
```




Terraform Basics and Essentials

Debug:

TF_LOG – Env variable to set log levels for terraform

- Available log levels - TRACE, DEBUG, INFO, WARN or ERROR

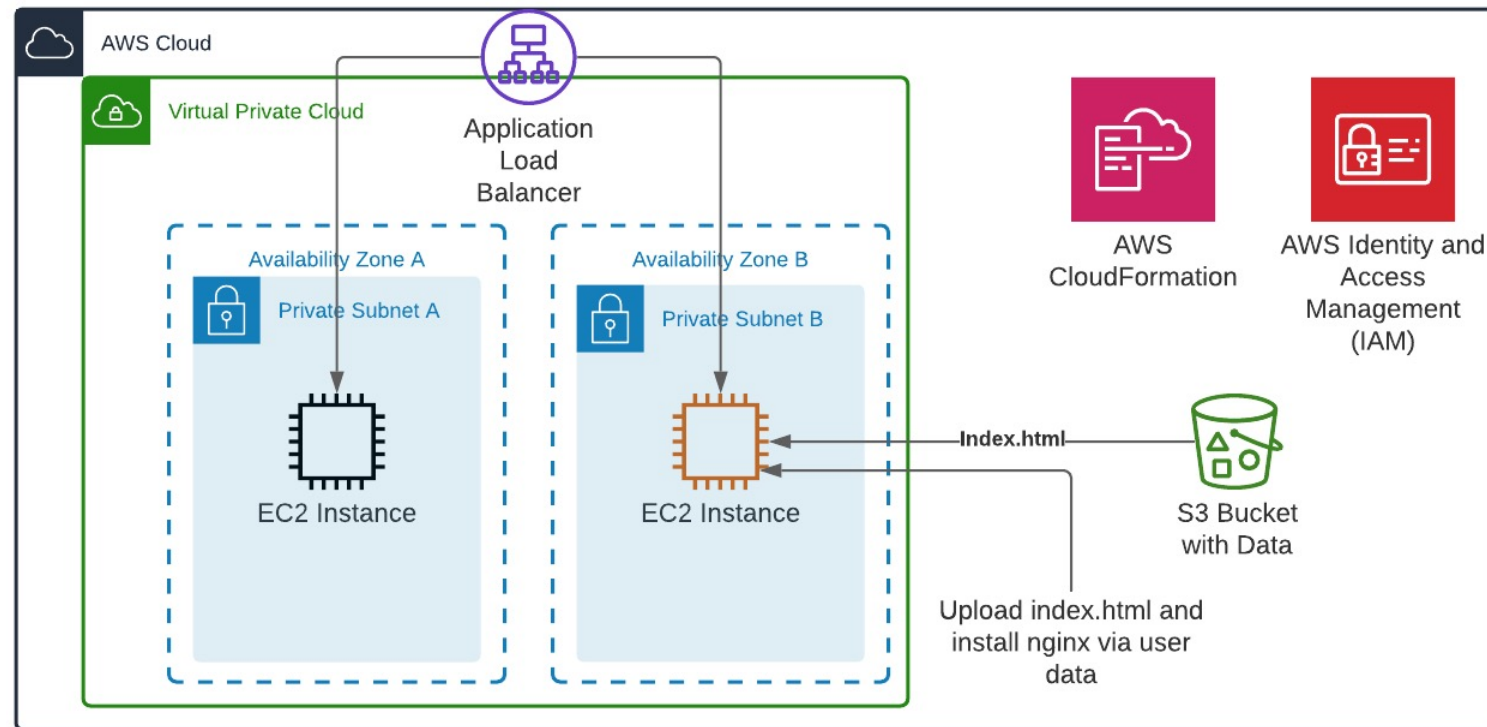
TF_LOG_PATH – Env variable to set path to persist log files

Опциональная домашка :

Написать *Terraform template* для диаграммы.

Доп. челлендж:

- Сделать тоже самое но в **Azure ARM** или **GCP Deploy Manager** и перерисовать диаграмму
- Для **AWS** обновить диаграмму, добавить упущенные сущности.
- Для **AWS EC2** добавить опцию **AutoRecovery**.
- Написать скрипт обертку и добавить возможность запускать разные *Envs* с разными параметрами.



Ref. Docs :

- <https://www.terraform.io/docs/providers/index.html>
- <https://www.terraform.io/docs/index.html>
- <https://registry.terraform.io/browse/providers>
- <https://www.terraform.io/docs/language/functions/index.html>
- https://learn.hashicorp.com/tutorials/terraform/troubleshooting-workflow?utm_source=WEBSITE&utm_medium=WEB_IO&utm_offer=ARTICLE_PAGE&utm_content=DOCS