

## Homework\_Lesson37\_IaC\_Terraform\_2

1. Создать конфигурацию для S3/GCP Storage. Добавить модуль, который будет отвечать за доп. параметры бакета (имя, размер, файлы для загрузки).
2. Используйте созданный модуль в основной конфигурации для развертывания инфраструктуры.
3. Настройте удаленное хранение tfstate в Terraform, используя AWS S3/GCP Storage в связке с AWS DynamoDB/CloudSQL
4. Перенесите свою существующую инфраструктуру на удаленное хранение стейта.
5. Создать в ручном режиме любой ресурс (лучше всего бакет либо VM) и импортируйте его в свою конфигурацию.
- 5.\* Создайте конфигурацию Terraform, в которой используется "depends\_on" для определения зависимостей между ресурсами. Убедитесь, что ресурсы создаются или обновляются в правильном порядке, используя это ключевое слово.

### Выполнение задания:

Перед выполнением создаем директорию, где будет создаваться наша конфигурация:

```
$ mkdir my-test-project
```

Так, как будем создавать модули (отдельные и переиспользуемые компоненты инфраструктуры), внутри созданной директории создаем доп. директорию "storage", в которой будем создавать конфигурацию модулей, в которых будут храниться файлы, описывающие ресурсы и логику конкретного модуля:

```
$ cd ./my-test-project          // в данной директории будет развертываться основная
                                // конфигурация инфраструктуры
$ mkdir module-gcp
$ cd ./module-gcp
$ mkdir storage                 // в данной директории будем создавать файл конфигурации
```

### 1. Создать конфигурацию для S3/GCP Storage. Добавить модуль, который будет отвечать за доп. параметры бакета (имя, размер, файлы для загрузки).

Создаем файл main-bucket.tf в директории /my-test-project/module-gcp/storage.

Добавляем скрипт описания модуля:

```
variable "bucket_name" {
  description = "Имя бакета"
  type        = string
}

variable "location" {
  description = "Местоположение бакета"
  type        = string
  default     = "US"
}

resource "google_storage_bucket" "bucket" {
  name        = var.bucket_name
  location    = var.location
  storage_class = "STANDARD"
}
```

```
output "bucket_name" {
  value = google_storage_bucket.bucket.name
}
```

## 2. Используйте созданный модуль в основной конфигурации для развертывания инфраструктуры.

В корне нашего проекта /my-test-project создаем файл main.tf:

```
terraform {
  required_version = "1.11.1"
  required_providers {
    google = {
      source = "hashicorp/google"
      version = "~>6.24.0"
    }
  }
}

provider "google" {
  credentials = file("/home/makarov/modified-glyph-450412-k6-a498aa55c17d.json")
  project = var.project_id
  region = var.region
}

module "storage" {
  source      = "./module-gcp/storage"
  bucket_name = var.bucket_name
}

variable "project_id" {
  description = "ID проекта GCP"
}

variable "region" {
  description = "Регион для развертывания ресурсов"
  default     = "us-central1"
}

variable "bucket_name" {
  description = "Имя бакета"
  default     = "my-unique-bucket-name"
}

output "storage_bucket_name" {
  value = module.storage.bucket_name
}
```

В нашей основной конфигурации (в корне проекта) используем нами созданный ранее модуль:

```
module "storage" {
  source      = "./module-gcp/storage"
  bucket_name = var.bucket_name
}
```

Также в нашей конфигурации мы используем переменные. Создаем файл terraform.tfvars, где хранятся наши данные переменных:

```
project_id = "modified-glyph-450412-k6"
bucket_name = "bucket-gcp-dz37"
region = "us-central1"
location = "US"
```

### 3,4 Настройте удаленное хранение tfstate в Terraform, используя AWS S3/GCP Storage в связке с AWS DynamoDB/CloudSQL. Перенесите свою существующую инфраструктуру на удаленное хранение стейта.

Для того, чтобы перенести нашу инфраструктуру на удаленное хранение стейта, добавляем в наш основной main.tf блок:

```
resource "google_storage_bucket_object" "file" {
  name     = "Hello_world_DOS24.txt"
  bucket   = module.storage.bucket_name
  content_type = "text/plain"
  source   = "/home/makarov/my-test-project/Hello_world_DOS24.txt"
}
```

Здесь мы переносим наш файл "Hello\_world\_DOS24.txt на удаленное хранение стейта.

После всех манипуляций запускаем инициализацию терраформа:

```
$ terraform init
$ terraform plan
$ terraform apply
```

```
makarov@DESKTOP-UG6J7T7:~/my-test-project$ terraform apply
module.storage.google_storage_bucket.bucket: Refreshing state... [id=bucket-gcp-dz37]
google_storage_bucket_object.file: Refreshing state... [id=bucket-gcp-dz37-Hello_world_DOS24.txt]

Note: Objects have changed outside of Terraform

Terraform detected the following changes made outside of Terraform since the last "terraform apply" which may have affected this plan:

# module.storage.google_storage_bucket.bucket has been deleted
- resource "google_storage_bucket" "bucket" {
  id           = "bucket-gcp-dz37"
  name         = "bucket-gcp-dz37" -> null
  # (17 unchanged attributes hidden)

  # (2 unchanged blocks hidden)
}
```

```
Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

Enter a value: yes

module.storage.google_storage_bucket.bucket: Creating...
module.storage.google_storage_bucket.bucket: Still creating... [10s elapsed]
module.storage.google_storage_bucket.bucket: Creation complete after 13s [id=bucket-gcp-dz37]
google_storage_bucket_object.file: Creating...
google_storage_bucket_object.file: Creation complete after 0s [id=bucket-gcp-dz37-Hello_world_DOS24.txt]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.

Outputs:

storage_bucket_name = "bucket-gcp-dz37"
makarov@DESKTOP-UG6J7T7:~/my-test-project$
```

Наш bucket создан успешно с использованием модулей

Для проверки заходим через GUI на GCP:

←

Bucket details

Go to path

Refresh

bucket-gcp-dz37

Locationus (multiple regions in United States)Storage classStandardPublic accessSubject to object ACLsProtectionSoft Delete

Objects

Configuration

Permissions

Protection

Lifecycle

Observability

New

Inventory Reports

Folder browser

bucket-gcp-dz37

Buckets > bucket-gcp-dz37

Create folder

Upload

Transfer data

Filter by name prefix only

Filter

Show

Live objects only

<input type="checkbox"/>	Name	Size	Type	Created
<input type="checkbox"/>	Hello_world_DOS24.txt	22 B	text/plain	May 5

Наш созданный “bucket-gcp-dz37” с залитым нашим файлом “Hello\_world\_DOS24.txt”