

# Auto Hosting

Hosting Automatizado  
con Ansible y Terraform



ANSIBLE



HashiCorp

**Terraform**

Proyecto de fin de ciclo

CFGS ASIR

Francisco José Romero Morillo

Junio 2019

# Índice

## 1. Introducción

## 2. Desarrollo

2.1. Roles de Ansible

2.2. Terraform

2.3. Aplicaciones web

- Aplicación del Hosting

- Aplicación del panel de control

## 3. Demo

# 1. Introducción

**Objetivo:** hosting automatizado sin administración.

**Tecnologías:**

**Ansible:** gestión de la configuración.

**Terraform:** gestión de la infraestructura.

**Escenario:**

**Servidor principal:** creación de servidores.

**Servidores:** creación de páginas web.

## 2. Desarrollo

### Roles Ansible

#### Roles: Panel de Control

- Rol *common*  
Se encarga de instalar los paquetes necesarios.
- Rol *install\_panel*  
Se encarga de la instalación y puesta en funcionamiento del panel de control.

## Roles Ansible

### Roles: CMS

- Rol *hosting\_cms*  
Se encarga preparar el servidor para la instalación del cms.
- Rol *install\_cms*  
Se encarga de la instalación y puesta en funcionamiento del cms.

# Terraform

## Funcionamiento:

1. Fichero main.tf
2. Ficheros de estados.
3. AMI's y recursos.

## Amazon AWS:

1. ¿Qué es?
2. Uso en el proyecto.

## Terraform en el proyecto:

1. Configuración.

# Aplicaciones Web

## Aplicación: Hosting

Registro de Usuarios

Inicio de Sesión

Perfil de Usuario

Crear Servidores

Desconexión

Páginas de Error

# Aplicación: Hosting

## **Registro de usuarios:**

1. Cifrado con *hashlib*:

```
h = hashlib.new("md5",b"cadena")
```

```
h = hashlib.md5(b"cadena")
```

2. Inserción en bdd mysql.

## **Inicio de sesión:**

1. Búsqueda de credenciales en bdd mysql.
2. Comparación de credenciales.
3. Sesión con *bottle\_session*.

## **Perfil de usuario:**

Administración de servidores:

1. Búsqueda de los servidores en la bbdd mysql.
2. Muestra los servidores con dos opciones:
  - Administrar servidor.
  - Dar de baja.

## **Crear servidores:**

1. Guardar datos del servidor en fichero main.tf.
2. Creación del servidor con terraform.
3. Configuración del servidor con ansible.
4. Inserción del servidor en bbdd mysql y en el dns.

## Desconexión:

1. Borrar sesión con *bottle\_session*.

## Páginas de error:

Accedemos a ellas si se dan uno de estos dos casos:

1. Creación de un usuario con un dni existente.
2. Creación de un servidor con un nombre existente.

# Aplicaciones Web

## Aplicación: Panel

Inicio de Sesión

Instalación de CMS

Visualización de Logs

Desconexión

# Aplicación: Panel

## **Inicio de sesión:**

1. Búsqueda de credenciales en fichero del servidor.
2. Comparación de credenciales.
3. Sesión con *bottle\_session*.

## **Instalación de CMS:**

1. Guardar datos del servidor en fichero de variables.
2. Instalación del CMS con ansible.

## **Visualización de logs:**

1. Lectura de logs del sistema.
2. Impresión de los logs por pantalla.

## **Desconexión:**

1. Borrar sesión con *bottle\_session*.

*Demo*