

# Automatización con Terraform para WordPress de Alta Disponibilidad en AWS

Entiendo que necesitas automatizar la implementación de WordPress de alta disponibilidad en AWS para una competición. Voy a crear un código Terraform basado en la actividad que has realizado manualmente, para que puedas reducir significativamente el tiempo de despliegue.

Analizando tu documento, veo que necesitas implementar:

- VPC con subredes públicas y privadas
- Grupos de seguridad
- EFS para almacenamiento compartido
- RDS Multi-AZ con MySQL
- Application Load Balancer con SSL
- Auto Scaling con WordPress

## Explicación del Código Terraform

He creado un archivo Terraform completo que implementa toda la infraestructura necesaria para un WordPress de alta disponibilidad en AWS. Aquí está lo que incluye:

1. **VPC y redes:**
  - VPC con CIDR 10.0.0.0/16
  - Subredes públicas y privadas en 2 zonas de disponibilidad
  - Internet Gateway y NAT Gateways para acceso a internet
2. **Grupos de seguridad:**
  - ALB: para el balanceador (puertos 80 y 443)
  - Webserver: para las instancias EC2 de WordPress
  - RDS: para la base de datos MySQL
  - EFS: para el sistema de archivos compartido
3. **Almacenamiento:**
  - Sistema de archivos EFS para compartir contenido entre instancias
  - Puntos de montaje en cada zona de disponibilidad
4. **Base de datos:**
  - Instancia RDS MySQL Multi-AZ para alta disponibilidad

- Grupo de subredes para RDS en las subredes privadas
- 5. **Balanceo de carga:**
  - Application Load Balancer público
  - Target Group para las instancias EC2
  - Certificado SSL mediante ACM
  - Redirección de HTTP a HTTPS
- 6. **Auto Scaling:**
  - Launch Template con script de inicialización para WordPress
  - Auto Scaling Group con mínimo 2, máximo 4 instancias
  - Políticas de escalado basadas en CPU
- 7. **Monitorización:**
  - Alarmas de CloudWatch para CPU alta/baja
  - Alarma para errores HTTP 5XX
- 8. **Bucket S3 para backups**

## Cómo usar este código

1. **Crea un archivo de variables:** Crea un archivo `terraform.tfvars` con los siguientes valores:

```
hcl
aws_region = "us-east-1" # Cambia según tu región preferida
db_password = "TuContraseñaSegura"
domain_name = "tudominio.com" # Dominio para el certificado SSL
```

2. **Ajusta la AMI y la clave SSH:** En el recurso `aws_launch_template`, asegúrate de:
  - Actualizar `image_id` según la región que uses (AMI Ubuntu)
  - Cambiar `key_name` por el nombre de tu par de claves SSH
3. **Inicializa y aplica:**

```
bash
terraform init
terraform apply
```

## Aspectos manuales que requieren atención

Estos son los pasos que aún necesitarás realizar manualmente:

1. **Validación del certificado SSL:** Deberás validar el certificado SSL añadiendo los registros DNS correspondientes.
2. **Configuración del dominio:** Tras el despliegue, necesitarás crear un registro CNAME o A para dirigir tu dominio al ALB.

3. **Configuración inicial de WordPress:** La primera vez que accedas, tendrás que completar el asistente de configuración de WordPress.
4. **Scripts para backups:** Aunque el bucket S3 se crea automáticamente, tendrás que configurar manualmente los scripts de backup de la base de datos.

Este código te permitirá implementar rápidamente la infraestructura de WordPress escalable y de alta disponibilidad para tu competición, reduciendo significativamente el tiempo de implementación comparado con el proceso manual.