



## Infraestructura II

# Crear un servidor de WordPress con MySQL

¡Bienvenido! En este espacio vamos a poner en práctica todo lo que aprendimos durante esta semana. ¡Vamos!

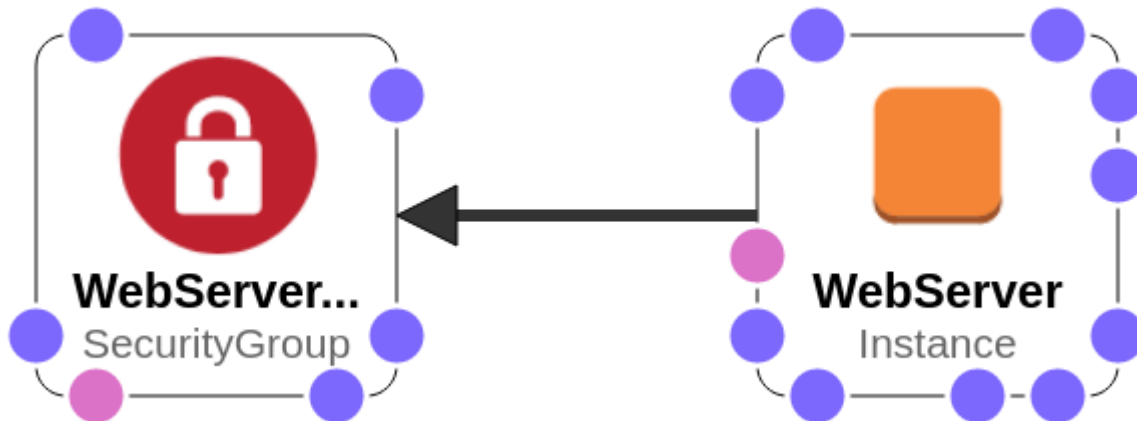
## Objetivo

Vamos a usar nuestra cuenta de AWS y utilizar un template de CloudFormation para crear un servidor de WordPress con su base de datos local, la misma se podrá visualizar desde internet a través de una IP Pública. Tu desafío va a ser llegar al final de este ejercicio.

## Instrucciones

Esta tarea es algo normal en el área de infraestructura de cualquier empresa. En nuestro caso ya lo tenemos automatizado por lo que con los permisos necesarios, vamos a poder ejecutarlo nosotros mismos a medida!.

El tipo de infraestructura que vamos a crear se diagrama de la siguiente manera:



El mismo consta de un web server que va a contener nuestra aplicación de WordPress con su base de datos de MySQL.

La única tarea previa que necesitamos del área de seguridad o bien de infraestructura, es un par de llaves para ingresar al servidor, pero en nuestro caso lo vamos a hacer nosotros. ¡Comencemos!

Al ingresar a nuestra cuenta de AWS, en el cuadro de búsqueda ingresamos "EC2". En el menú de la izquierda dentro de la sección "Red y Seguridad" y seleccionamos "Par de Llaves". Vamos a ingresar un nombre para nuestras llaves, por ejemplo "wordpress-access" y lo guardamos en nuestro equipo. Es muy importante que no se extravíen para no perder acceso a nuestro servidor, ¡guardalo muy bien!.



EC2 > Pares de claves > Crear par de claves

## Crear par de claves

### Par de claves

Un par de claves, compuesto por una clave privada y una clave pública, es un conjunto de credenciales de seguridad que se utilizan para demostrar su identidad cuando se conecta a una instancia.

**Nombre**

El nombre puede incluir hasta 255 caracteres ASCII. No puede incluir espacios al principio ni al final.

**Private key file format**

☒ .pem  
Para usar con OpenSSH

☐ .ppk  
Para usar con PuTTY

**Etiquetas (opcional)**

No hay etiquetas asociadas a este recurso.

Puede agregar 50 etiquetas más.

Cancelar

Ahora sí vamos a poner manos a la obra. Vamos usar el siguiente link para usar un template previamente creado por el equipo de AWS:

[https://docs.aws.amazon.com/es\\_es/AWSCloudFormation/latest/UserGuide/sample-templates-applications-us-west-1.html](https://docs.aws.amazon.com/es_es/AWSCloudFormation/latest/UserGuide/sample-templates-applications-us-west-1.html)

Vamos a elegir nuestro ejemplo para ejecutarlo haciendo click en “Launch Wizard”, esto nos va a redirigir a nuestra cuenta de AWS, directamente al servicio CloudFormation.



AWS > Documentación > AWS CloudFormation > Guía del usuario

Region

- US East (Northern Virginia) Region
- US East (Ohio) Region
- US West (Northern California) Region

**Soluciones de ejemplo**

- Marcos de aplicaciones
- Servicios
- US West (Oregon) Region

Solución de problemas

Historial de versiones

AWS glossary

Las plantillas de soluciones de ejemplo le muestran cómo crear una solución integral con aplicaciones comunes. AWS no admite ni mantiene las aplicaciones de estos ejemplos y su objeto es demostrar las capacidades de plantillas de AWS CloudFormation.

Nombre de la plantilla	Descripción	Ver	Ver en Designer	Lanzar
Windows Server				
Instancia individual básica de WordPress	Instala e implementa WordPress en una sola instancia Amazon EC2 con una base de datos MySQL local para el almacenamiento.	Ver	Ver en Designer	Launch stack
Escalable y duradera de	Instala e implementa WordPress en instancias Amazon EC2 en un grupo de	Ver	Ver en Designer	Launch stack

La información del template ya se encuentra cargada, únicamente tenemos que elegir "Next" y continuar con los pasos siguientes.

CloudFormation > Stacks > Create stack

**Step 1: Specify template**

**Step 2: Specify stack details**

**Step 3: Configure stack options**

**Step 4: Review**

**Create stack**

**Prerequisite - Prepare template**

**Prepare template**  
Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☒ Template is ready ☐ Use a sample template ☐ Create template in Designer

**Specify template**  
A template is a JSON or YAML file that describes your stack's resources and properties.

**Template source**  
Selecting a template generates an Amazon S3 URL where it will be stored.

☒ Amazon S3 URL ☐ Upload a template file

Amazon S3 URL

Amazon S3 template URL  
S3 URL: https://s3.us-west-1.amazonaws.com/cloudformation-templates-us-west-1/WordPress\_Single\_Instance.template

Como nuestra aplicación es solo un ejemplo para comprobar el potencial de esta herramienta, solo vamos a completar el Paso 2.



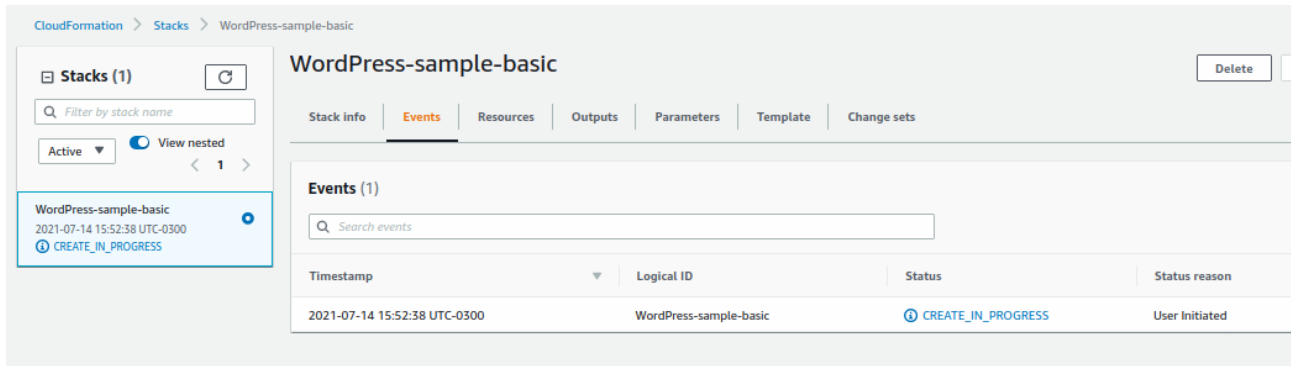
The screenshot shows the AWS CloudFormation console interface. On the left, a sidebar lists the steps: Step 1: Specify template, Step 2: Specify stack details (active), Step 3: Configure stack options, and Step 4: Review. The main area is titled 'Specify stack details'. It contains a 'Stack name' field with the value 'WordPress-sample-basic'. Below this is a 'Parameters' section with several input fields: 'DBName' (wordpressdb), 'DBPassword' (masked), 'DBRootPassword' (masked), 'DBUser' (masked), 'InstanceType' (t2.small), 'KeyName' (wordpress-access), and 'SSHLocation' (0.0.0.0/0). At the bottom right of the form are three buttons: 'Cancel', 'Previous', and 'Next'.

Para el paso 3 y paso 4, podemos seleccionar “Next” para ir al final del formulario y hacer click en “Create Stack”

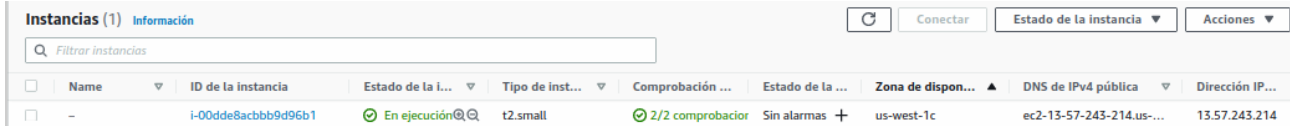
This image is a close-up of the bottom navigation bar of the AWS CloudFormation console. It contains four buttons: 'Cancel', 'Previous', 'Create change set', and 'Create stack'. The 'Create stack' button is highlighted in orange, indicating it is the primary action.

Los invitamos a investigar las opciones y que lo ejecuten varias veces con distintas combinaciones de opciones. Siempre tenemos la documentación oficial de AWS para seguir aprendiendo.

Estamos en el final listos para ver cómo se ejecuta!



Al finalizar, vamos a poder visualizar el mensaje “Create Complete”. Volvamos a la sección de EC2 desde el cuadro de búsqueda de servicios, copiamos la IP Pública y vamos a poder acceder a nuestra instancia de WordPress, en el caso de estas capturas es la IP 13.57.243.214, aunque cada uno al momento de ejecutar esta práctica, AWS le asigna una IP distinta.



Name	ID de la instancia	Estado de la i...	Tipo de inst...	Comprobación ...	Estado de la ...	Zona de dispon...	DNS de IPv4 pública	Dirección IP...
-	i-00dde8acbb9d96b1	En ejecución	t2.small	2/2 comprobaci...	Sin alarmas	us-west-1c	ec2-13-57-243-214.us-...	13.57.243.214

## Repasemos los recursos que se crearon para la práctica:

- Un servidor virtual con:
  - Un webserver apache para exponer la aplicación hacía internet.
  - Una aplicación wordpress dentro del webserver.
  - Una base de datos MySQL local.
- Grupos de seguridad, en nuestro caso para acceder desde internet.
- Un grupo de seguridad asociado a nuestra instancia EC2
- Un volumen asignado a nuestro servidor, para que los datos persistan aunque se reinicie el servidor.

¡Todo esto con solo algunos clicks desde nuestro template! Pero si quiero eliminar estos recursos, ¿debo hacerlo uno por uno?



¡La respuesta es no! vamos a poder eliminarlos igual de fácil desde CloudFormation, únicamente eliminando la pila de nuestro WordPress.

