

# SONY ETCHEVERRY

sonyetchevery13@gmail.com | 093967511

21 AGOSTO DE 2024

AUTORES:

S.ETCHEVERRY - G.RONDEAU

## Implementación de un Servidor de Minecraft en AWS

### 1. Introducción

Este documento detalla el proceso de creación e implementación de un servidor de Minecraft utilizando la capa gratuita de Amazon Web Services (AWS), específicamente a través de Amazon EC2. El objetivo es evaluar la viabilidad de utilizar los recursos gratuitos de AWS para alojar un servidor de Minecraft accesible a través de internet.

### 2. Motivaciones

El propósito principal de este proyecto fue determinar si es factible configurar y operar un servidor de Minecraft sin incurrir en costos, utilizando exclusivamente los recursos disponibles en la capa gratuita de AWS. Este enfoque permite que cualquier usuario con acceso al nivel gratuito de AWS pueda replicar el proceso desde su propia ubicación.

### 3. Implementación Técnica

#### 3.1 Creación de EC2

Se inicia el proceso creando una instancia EC2 desde la consola de administración de AWS, utilizando una cuenta en el nivel gratuito (Free Tier).

Amazon EC2 proporciona una infraestructura escalable y flexible para ejecutar aplicaciones. La selección de una instancia dentro del Free Tier asegura que el proceso no incurra en costos adicionales durante la fase de prueba.

### 3.2 Conexión mediante SSH

Se utiliza la herramienta PuTTY para conectarse a la instancia EC2 y ejecutar comandos remotos.

La conexión SSH permite la administración segura y remota de la instancia, facilitando la configuración del servidor sin la necesidad de acceso físico a la máquina.

### 3.3 Instalación de Java

Se instala Java en la instancia para habilitar la ejecución del servidor de Minecraft.

Minecraft requiere Java para funcionar. La instalación de Java es esencial para que el servidor pueda ejecutar el software de Minecraft correctamente.

### 3.4 Creación y Ejecución de Script

Se desarrolla un script .sh para automatizar el inicio del servidor de Minecraft. Se asignan permisos de ejecución al script.

La automatización del proceso de inicio mediante un script simplifica la administración del servidor y asegura que el servidor se ejecute con la configuración deseada tras cada reinicio.

### 3.5 Edición de Archivo EULA

Se edita el archivo eula.txt para aceptar los términos y condiciones de Minecraft.

Aceptar el archivo eula.txt es un requisito legal para operar el servidor de Minecraft, asegurando el cumplimiento de los términos del servicio del juego.

### 3.6 Edición de Propiedades del Servidor

Se ajustan las propiedades del servidor, incluyendo la cantidad máxima de jugadores y el modo de juego.

La configuración de las propiedades del servidor permite personalizar la experiencia de juego y adaptar el servidor a los requisitos específicos del proyecto.

### 3.7 Lanzamiento del Servidor

Se inicia el servidor con las configuraciones ajustadas para verificar su funcionamiento.

El lanzamiento del servidor permite comprobar que todas las configuraciones sean correctas y que el servidor funcione como se espera.

### 3.8 Obtención de IP

Se obtiene la dirección IPv4 pública de la instancia EC2.

La dirección IP pública es necesaria para que los jugadores se conecten al servidor desde internet. Sin esta IP, el servidor no sería accesible desde fuera de la red local.

### 3.9 Dominio del Servidor

Se configura un dominio gratuito utilizando Duck DNS para gestionar la IP dinámica.

La IP asignada por AWS puede cambiar, lo que dificultaría la conexión al servidor. Un dominio fijo facilita el acceso constante al servidor sin necesidad de actualizar la IP manualmente.

### 3.10 Habilitación del Puerto de Minecraft

Se configura el grupo de seguridad de la instancia para permitir el acceso al puerto 25565, utilizado por Minecraft.

Minecraft requiere el puerto 25565 para las conexiones de red. La habilitación de este puerto es esencial para que los jugadores puedan conectarse al servidor.

### 3.11 Entrada al Servidor

Se agrega el servidor en el launcher de Minecraft utilizando el DNS configurado.

Ingresar el servidor en el launcher de Minecraft con el DNS facilita a los jugadores la conexión al servidor sin problemas adicionales.

### 3.12 Monitoreo del Servidor

Se monitorea el uso de recursos de la instancia utilizando el comando top.

El monitoreo de recursos es crucial para asegurar que la instancia no esté sobrecargada y que el servidor funcione de manera estable.

## 4. Problemáticas

Durante el proyecto se identificó que la instancia t2.micro, con 1 vCPU y 1 GB de RAM, presenta limitaciones significativas para manejar servidores más exigentes. Como solución, se optó por la versión 1.12.2 de Minecraft, la cual tiene menores requisitos de sistema y resultó ser más compatible con los recursos disponibles.

## 5. Consideraciones de Servicios Free Tier

El Free Tier de Amazon EC2 proporciona una micro instancia gratuita durante 12 meses, con un límite de 750 horas de uso mensual. Es crucial realizar un seguimiento del uso para evitar costos adicionales y considerar los gastos asociados con servicios complementarios, como almacenamiento adicional.

## 6. Alternativas

Una alternativa viable a EC2 es Amazon Lightsail, que ofrece más potencia de cómputo gratuita durante 3 meses. Esta opción podría ser más adecuada para ejecutar versiones recientes de Minecraft que requieren mayores recursos.

## 7. Conclusiones

El proyecto demostró la viabilidad de alojar un servidor de Minecraft utilizando los recursos gratuitos de AWS. A pesar de las limitaciones de la instancia t2.micro, se logró proporcionar una experiencia de juego adecuada. La implementación de un dominio gratuito con Duck DNS resolvió el problema de la IP dinámica, facilitando