

Cómputo en Nube

AWS: EC2, Lambda y otras herramientas



ALBAÑILESDIGITALES

Developing your new career



Hoy veremos

1. **Breve repaso** a la nube y al cómputo
2. **¿Qué vamos a estudiar nosotros?**
3. Diferentes maneras de tener **cómputo en AWS**
 - a. Solo código -> **Lambda**
 - b. Usando un servidor -> **EC2**
 - c. Usando varios servidores -> **ECS**
 - d. Ignorando servidores -> **Lightsail/Fargate**
4. Conclusiones

Breve repaso a la nube

- Pero espera... ¿Qué es el cómputo?
 - **Cómputo es cálculo**
 - Los ordenadores para funcionar hacen mucho cálculos matemáticos
 - **A más cómputo, más rápido irá el ordenador o servidor.**
- La nube nos permite **alquilar servidores**
- Esto es beneficioso si queremos ser **flexibles con nuestro cómputo**

Breve repaso a la nube

- Es como comprar un piso o alquilarlo: Alquilar te permite probar diferentes pisos, cambiar cada tiempo, etc. aunque a la larga acabes pagando más
 - Generalmente en informática queremos ser flexibles
 - Cada cierto tiempo hay **nuevas tecnologías que mejoran el cómputo** (y otras cosas)
 - Además a veces querremos más cómputo y otras menos
 - No necesitamos un desembolso inicial

Breve repaso al cómputo

- Pero entonces... ¿Hay diferentes formas de ejecutar cómputo?
 - **Si, y no.**
 - **Si** hay diferentes formas y son las que vamos a ver a continuación
 - **Y no** porque en realidad todo siguen siendo servidores, solo que AWS te ayuda a gestionarlo y te da otras utilidades

Breve repaso al cómputo

- ¿Y **por qué** hay diferentes formas?
 - Ayudar al desarrollador a conseguir su objetivo final más fácilmente
 - Alguien que quiera hacer su web y no preocuparse de nada más usará algo más simple
 - Sin embargo alguien que quiera más control sobre su infraestructura usará herramientas más complejas

¿Qué vamos a estudiar nosotros?

- Vamos a intentar tocar un poco de todo:
- Primero haremos **un ejercicio más básico: ejecutar código**. Seguramente nos cueste un rato, pero luego lo haremos en otra herramienta más simple en pocos segundos. Así entenderemos por qué hay tantas herramientas
- **La siguiente tarea será la principal** y con la que estaremos todo el bloque: **Mantener un servicio API en la nube**.
 - Primero veremos cómo ejecutarlo de forma básica
 - Luego nos centraremos en ver cómo lo hacemos más potente
 - Finalmente veremos cómo gestionarlo y mantenerlo en el tiempo

¿Un servicio API?

- Recordemos que es un **servicio API**:
 - Servicio API = **Servicio que expone una API**
 - API = Application Program Interface. Es decir, **un protocolo de comunicación para apps**
 - Una serie de **métodos** de un programa **pensados para ser ejecutados desde otro programa**
 - En una estructura Cliente/Servidor, **las API's estarían en el servidor.**
 - El cliente realizaría las consultas

¿Un servicio API?

- **Antes muchísima carga se hacía en el servidor.**
 - Se creaba la web en el servidor y se enviaba
 - El navegador la visualizaba
- **Ahora con las API's esto se separa.**
 - **FrontEnd por un lado** -> Se encarga de la visualización
 - **BackEnd por el otro** -> Se encarga del cómputo
- Se establecen estructuras de datos para la comunicación
 - JSON
 - XML

¿Un servicio API?

- Ejemplo: [Pokeapi](https://pokeapi.co/api/v2/pokemon/pikachu)
 - Url: <https://pokeapi.co/api/v2/pokemon/pikachu>
 - Https -> esquema
 - Pokeapi.co -> host
 - /api/v2 -> prefijo
 - /pokemon/pikachu -> recurso
- Aunque realmente esto no nos hace falta ahora mismo, ya lo iremos viendo en la parte práctica.

Cómputo en AWS

- Hemos dicho que hay diferentes maneras de ejecutar cómputo en AWS:
- Las más famosas y las que vamos a estudiar son:
 - **AWS Lambda**
 - **EC2**
 - **ECS**
 - **Lightsail/Fargate**
- Vamos a estudiar una por una para que sirven

AWS Lambda

- La forma más **fácil** de ejecutar cómputo en la nube
- Sólo tienes que poner el código y te despreocupas al 100% del resto
- Pros:
 - Muy muy fácil de usar
 - **No te preocupas por cómo desplegar**, solo por el código
 - ¡Muy **barato**!
 - No te preocupas por cuanto cómputo necesitas. **Escalado totalmente automático.**
 - No tienes que ocuparte de los servidores

AWS Lambda

- La forma más **fácil** de ejecutar cómputo en la nube
- Sólo tienes que poner el código y te despreocupas al 100% del resto
- Contrás:
 - Estás **atado a AWS**, tu solución sólo existe ahí
 - Si quieres hacer algo **complicado**, es aún más complicado. Esto se debe a que estás obligado a usar herramientas de AWS
 - **Cold Start**: Tarda unos segundos en ejecutarse.
- Dependiendo lo que hagas... puede que no tenga sentido. **No es una solución para todos los casos.**

EC2

- Un **servidor de toda la vida** pero alquilado
- Pros:
 - **Flexibilidad** en cuanto uso de Sistema Operativo y programas instalados
 - Mayor flexibilidad en cpu/memoria (cómputo)
 - Mayor **control** sobre el entorno de configuración
 - En general... mayor control y flexibilidad

EC2

- Un **servidor de toda la vida** pero alquilado
- Contras:
 - El mayor control/flexibilidad tiene un precio, y es que todo será **más complicado** de configurar
 - Muchas de las cosas que te da lambda hechas, las tienes que hacer tú
 - **Mayor costo.** Una lambda se ejecuta el tiempo que el código funcione, la instancia EC2 es 24/7

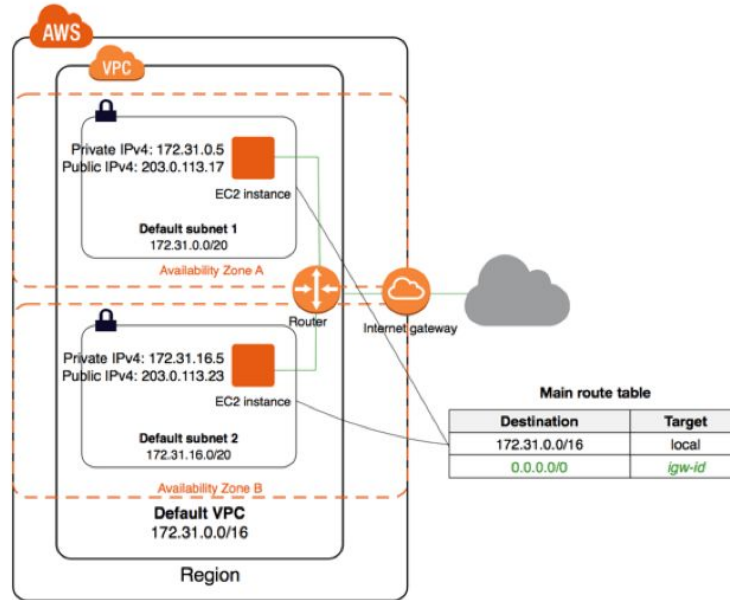
Un poco de networking



VPC

- Virtual Private Cloud → Redes virtuales en la nube
- Puedes configurar todos los elementos de una red:
 - Subredes
 - Internet Gateways
 - Tablas de rutas
 - Interfaces de red

VPC



EC2 – ELB

- Elastic Load Balancing: balanceadores de carga
- Los hay de capa de red (NLB) y de capa de aplicación (ALB)
- Pueden apuntar a destinos variados
 - Instancias
 - Direcciones IP
 - Otros balanceadores...

ECS

- Sirve para organizar **varias máquinas haciendo el mismo trabajo**
- En vez de tener un servidor potente, tienes varios menos potentes
- En las máquinas habrá **contenedores** ejecutándose
- Pros:
 - Eres más **flexible**. Al tener más máquinas, puedes decidir quitar y poner según haga falta y pagar menos
 - Te aseguras que si se cae la máquina, hay **otra funcionando**
 - Es una de las herramientas más **fáciles** para hacer esta tarea (kubernetes suele ser la alternativa, mucho más compleja)

ECS

- Sirve para organizar **varias máquinas haciendo el mismo trabajo**
- En vez de tener un servidor potente, tienes varios menos potentes
- En las máquinas habrá **contenedores** ejecutándose
- Contrás:
 - Añade otra **capa de complejidad**. Más herramientas y más cosas a configurar
 - Tienes que **usarlo junto a EC2**.
 - Al haber varias máquinas trabajando haciendo lo mismo, se tienen que comunicar entre ellas. **Trabajo extra**.

Lightsail/Fargate

- Dos herramientas diferentes que **ayudan a ejecutar cómputo sin preocuparnos de los servidores**
- Son **puntos intermedios** a AWS Lambda o EC2
- **Fargate:**
 - Te permite **lanzar contenedores** en máquinas sin gestionar los servidores.
 - Esto te permite **usar ECS sin necesidad de usar una EC2**
 - Seguramente una de las soluciones más recomendadas si no necesitamos un 100% de flexibilidad

Lightsail/Fargate

- Dos herramientas diferentes que **ayudan a ejecutar cómputo sin preocuparnos de los servidores**
- Son **puntos intermedios** entre lambda y ec2
- **Lightsail:**
 - Básicamente es **un PaaS** (Platform as a Service)
 - Uno se centra en desplegar su programa y se olvida del resto.
 - **Fácil de usar, pero muy poco configurable**
 - Esto lo hace menos recomendable, ya que a nada que quieras hacer algo diferente no podrás

Conclusiones Cómputo

- AWS ofrece **diferentes servicios** para ejecutar código **según la circunstancia**. Vamos a verlo de más simple a más complejo:
 - **AWS Lambda**: Nos olvidamos de todo, ponemos **código** y a funcionar
 - **EC2**: Levantas un **servidor** y lo configuras
 - **ECS**: Tenemos **varios servidores** y necesitamos que trabajen conjuntamente
- Además ofrece 2 herramientas extras que ofrecen puntos intermedios:
 - **Fargate**: Quiero **ECS sin usar EC2**
 - **Lightsail**: Quiero hacer **algo más que una lambda, pero sin ninguna complicación** más

Conclusiones

- AWS ofrece **diferentes servicios** para ejecutar código **según la circunstancia**
- Nosotros ahora nos vamos a **centrar en la parte práctica**, estudiando y descubriendo estas herramientas
- **Hoy veremos 2** de ellas: EC2 y AWS Lambda
- En los siguientes días **iremos estudiando otras** y viendo como funcionan entre ellas



ALBAÑILES DIGITALES

Developing your new career

