

Criterio	EC2	AWS Lambda	Beanstalk
Tipo Servicio	Infraestructura como servicio (IaaS)	Función como servicio (FaaS / Serverless)	Plataforma como servicio (PaaS)
Costo	Se paga por hora o segundo de ejecución de la instancia (según tipo y tamaño). También por almacenamiento, tráfico y otros servicios asociados.	Se paga por cada invocación y tiempo de ejecución en milisegundos. Muy económico para cargas pequeñas o esporádicas.	Se paga por los recursos subyacentes que usa (EC2, S3, RDS, etc.), pero la gestión es gratuita.
Casos de uso	Aplicaciones que requieren control total del entorno, servidores persistentes, o configuraciones personalizadas (bases de datos, servidores web, etc.).	Microservicios, tareas automatizadas, APIs ligeras, procesamiento por eventos (por ejemplo: S3, DynamoDB, CloudWatch).	Aplicaciones web completas en frameworks como Node.js, Python, Java, .NET, PHP, Go, con despliegue automatizado.
Gestión del servidor	Totalmente manual. El usuario debe actualizar el SO, parches, seguridad, etc.	No hay gestión de servidores; AWS lo maneja completamente.	AWS gestiona automáticamente servidores, balanceadores y escaladores.
Tiempo de arranque	Lento (minutos) al iniciar una nueva instancia.	Muy rápido (milisegundos) aunque puede tener "cold start" en algunas funciones.	Medio (depende del despliegue de EC2).
Ventajas principales	<ul style="list-style-type: none"> - Control total sobre hardware y SO. - Flexibilidad máxima. - Soporta cualquier stack. 	<ul style="list-style-type: none"> - Escalabilidad automática sin configuración. - Bajo costo para cargas variables. - Sin mantenimiento de servidores. 	<ul style="list-style-type: none"> - Despliegue rápido y automatizado. - Escalado y balanceo integrados. - Compatible con varios lenguajes.
Desventajas	<ul style="list-style-type: none"> - Requiere mantenimiento 	<ul style="list-style-type: none"> - Limitado por tiempo máximo de 	<ul style="list-style-type: none"> - Menos control sobre la

	<p>constante. - Puede ser costoso si no se optimiza. - Escalabilidad más lenta.</p>	<p>ejecución (15 min). - Difícil para procesos de larga duración. - Cold starts en algunas regiones.</p>	<p>infraestructura. - Menor personalización. - Puede ser más caro para grandes cargas constantes.</p>
Integración con otros servicios	<p>Alta (EC2 es base para muchos otros servicios).</p>	<p>Altísima: se integra nativamente con S3, DynamoDB, API Gateway, EventBridge, etc.</p>	<p>Alta: integra RDS, CloudWatch, S3, EC2, etc.</p>
Cuando usarlo	<p>Cuando necesitas control total del entorno, aplicaciones grandes o bases de datos dedicadas.</p>	<p>Cuando necesitas ejecutar tareas por evento o servicios ligeros con alta escalabilidad.</p>	<p>Cuando quieres desplegar una app web sin preocuparte por la infraestructura, pero con control moderado.</p>
Fortalezas	<p>Flexibilidad, rendimiento, control, opciones de red y seguridad avanzadas.</p>	<p>Escalado automático, simplicidad, costo por uso, ideal para microservicios.</p>	<p>Despliegue automatizado de aplicaciones completas sin gestión de servidores.</p>
Debilidades	<p>Administración manual, mantenimiento costoso, riesgo de sobreprovisión.</p>	<p>Limitaciones de tiempo y memoria, no apto para procesos persistentes.</p>	<p>Dependencia de EC2 y menor control sobre configuraciones finas.</p>