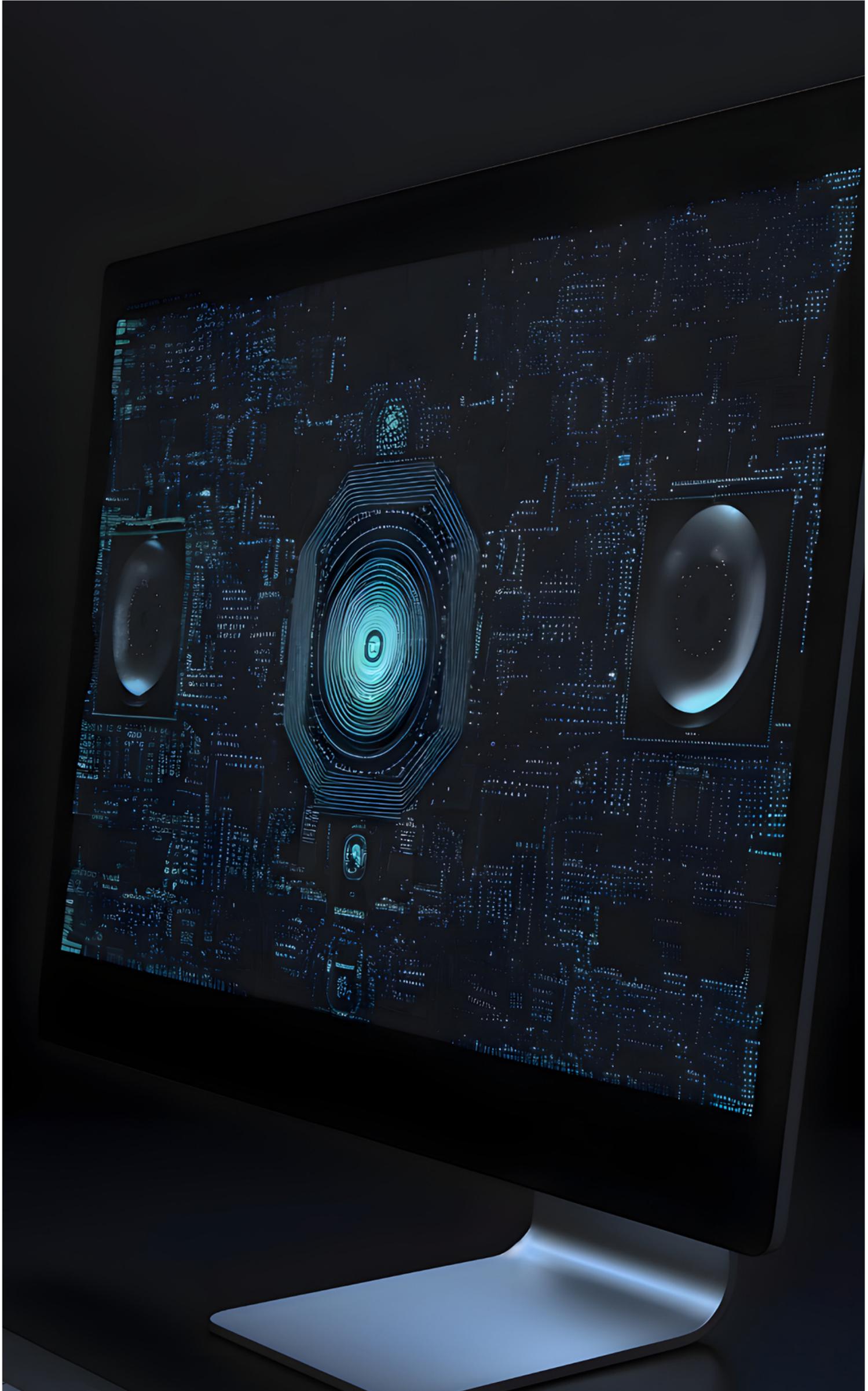


Lección 5

AWS Elastic Beanstalk



Desafíos generales

-  Administrar la infraestructura en torno a la implementación de aplicaciones puede ser *difícil*
-  Administrar y configurar servidores puede llevar *mucho tiempo*
-  Podría haber *falta de coherencia* entre varios proyectos o aplicaciones

Ahora, considere algunos desafíos generales que podría enfrentar a la hora de comenzar con la administración de la infraestructura de su nube.

En primer lugar, tenga en cuenta que implementar una aplicación puede ser difícil. ¿Cómo se asegurará de que está altamente disponible y que puede admitir las solicitudes de los usuarios incluso durante periodos de máximo uso? ¿Cómo puede asegurarse de que la aplicación es resistente y que se realizan copias de seguridad regulares con fines de recuperación de desastres (DR)? Administrar y configurar servidores puede llevar mucho tiempo. Mientras tanto, querrá mantener la coherencia entre proyectos y aplicaciones, pero dicha coherencia puede ser difícil de lograr.

AWS Elastic Beanstalk



AWS Elastic
Beanstalk

Una forma sencilla de poner en marcha las aplicaciones web
Un servicio administrado que gestiona automáticamente:

- Aprovisionamiento
- Implementación
- Balanceo de cargas
- Escalado automático
- Monitoreo de estado
- Análisis y depuración
- Registros

No hay cargo adicional por uso

- Pague solo por los recursos subyacentes que se utilizan



AWS Elastic Beanstalk es otra opción de servicio informático de AWS. Se trata de una plataforma como servicio (PaaS) que facilita la implementación, el escalado y la administración de aplicaciones y servicios web con rapidez. Aborda muchos de los desafíos que acaba de ver.

Con Elastic Beanstalk, usted mantiene el control de su código, mientras que AWS mantiene la infraestructura subyacente. Los recursos necesarios de AWS se crean e implementan mediante un simple asistente de la consola de administración de AWS. El asistente le pide que elija el tipo y el tamaño de la instancia, el tipo y el tamaño de la base de datos y la configuración de escalado automático que desea utilizar. Proporciona acceso a los archivos de registro del servidor y habilita HTTP seguro (HTTPS) en el balanceador de carga.

Solo tiene que cargar el código y Elastic Beanstalk administrará de manera automática la implementación, incluidos el aprovisionamiento de la capacidad, el balanceo de carga, el escalado automático y el monitoreo del estado de la aplicación. Al mismo tiempo, tendrá el control absoluto de los recursos de AWS que utiliza la aplicación y podrá acceder a los recursos subyacentes en cualquier momento.



No se aplican cargos adicionales por el uso de AWS Elastic Beanstalk. Solo tiene que pagar por los recursos de AWS que crea para almacenar y ejecutar la aplicación, como instancias EC2 o buckets de S3. Solo se paga por lo que se utiliza, cuando lo utiliza.

Implementaciones de AWS Elastic Beanstalk

- Admite aplicaciones web escritas para plataformas comunes

- Java, .NET, PHP, Node.js, Python, Ruby, Go, y Docker

- Usted carga el código.

- Elastic Beanstalk administra automáticamente la implementación
 - Se implementa en servidores como Apache, NGINX, Passenger, Puma y Microsoft Internet Information Services (IIS)



Elastic Beanstalk configura cada instancia EC2 del entorno con los componentes necesarios para ejecutar las aplicaciones de la plataforma seleccionada. Ya no tendrá que preocuparse por iniciar sesión en las instancias para instalar y configurar su pila de aplicaciones.

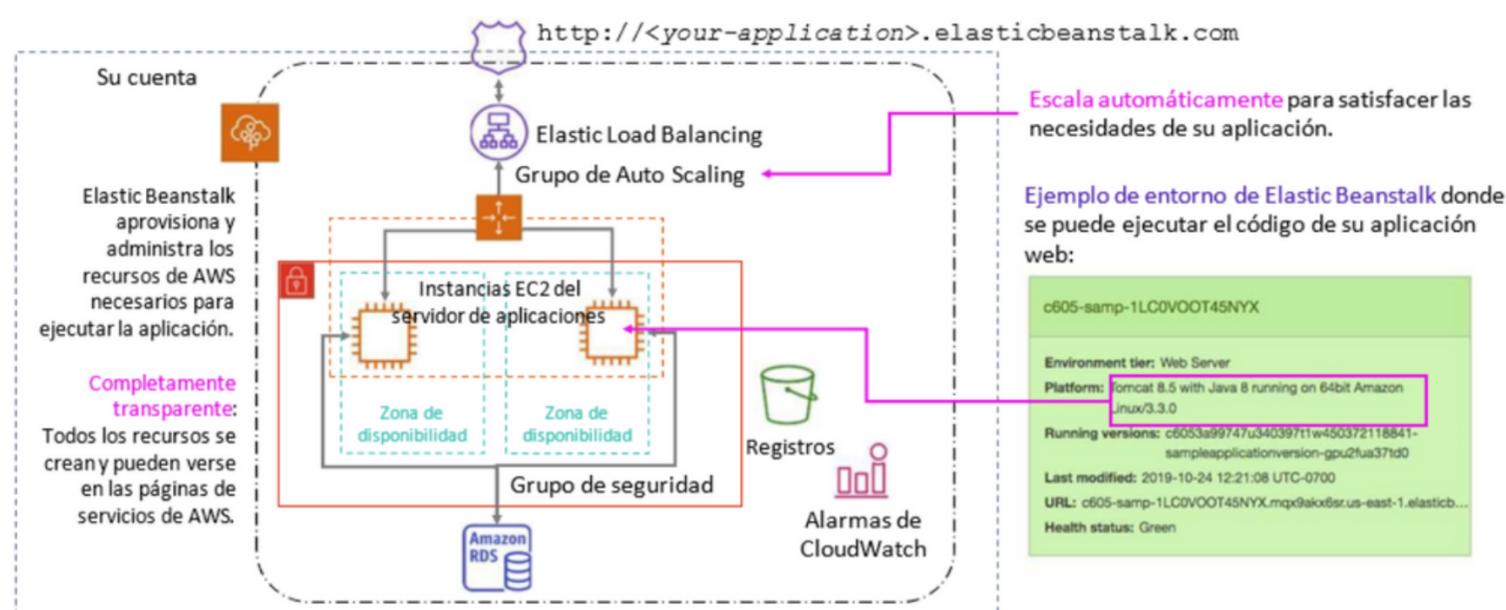
Lo único que debe crear es el código. Elastic Beanstalk está diseñado para que la implementación de su aplicación sea un proceso rápido y sencillo. Es compatible con una gama de plataformas, incluyendo Docker, Go, Java, .NET, Node.js, PHP, Python y Ruby.



AWS Elastic Beanstalk implementa el código en:

- Apache Tomcat para aplicaciones Java
- Servidor HTTP Apache para aplicaciones Python y PHP
- Servidor HTTP Apache o NGINX para aplicaciones Node.js
- Passenger o Puma para aplicaciones Ruby
- Microsoft Internet Information Services (IIS) para aplicaciones .NET, Java SE, Docker y Go.

Entorno de aplicaciones de Elastic Beanstalk



Cuando trabaja con Elastic Beanstalk, puede elegir entre dos tipos de entornos. El entorno de instancia única le permite lanzar una única instancia EC2 y no incluye balanceo de carga ni escalado automático. El otro tipo de entorno, que se encuentra en este ejemplo, puede lanzar varias instancias EC2 e incluye balanceo de carga y una configuración de escalado automático. Una capa de base de datos administrada es opcional.

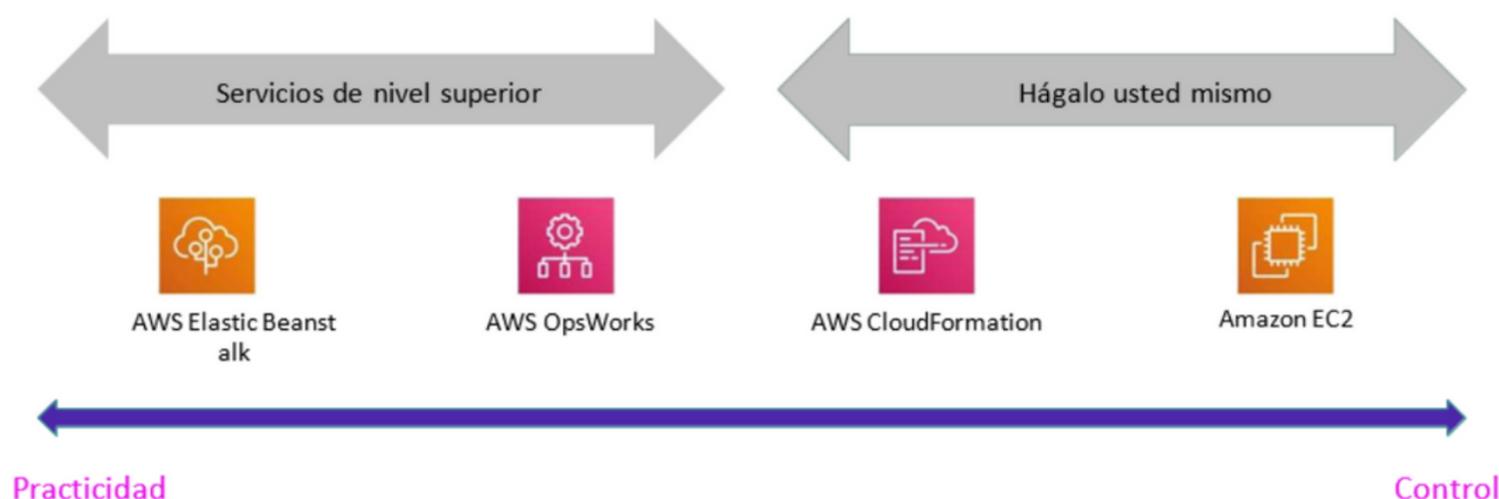
Los recursos de AWS creados por Elastic Beanstalk son visibles desde su cuenta de AWS. Por ejemplo, después de crear una aplicación de Elastic Beanstalk, abra la consola de administración de AWS y, a continuación, abra la consola de Amazon EC2. Verá las instancias que Elastic Beanstalk está administrando por usted. En el ejemplo, se crearon dos instancias EC2. Cada instancia EC2 ejecuta el sistema operativo invitado Amazon Linux, con Java 8 y el servidor web Apache Tomcat 8.5 instalados.



El código de la aplicación se ejecutará en estos servidores. El escalado automático está configurado, por lo que si la carga comienza a sobrecargar los recursos en estas dos instancias (como el uso excesivo de CPU durante más de 5 minutos), se lanzarán automáticamente más instancias del servidor de aplicaciones. En este ejemplo también se muestra una instancia de base de datos RDS que está disponible y es accesible desde las instancias EC2. Puede almacenar los datos de la aplicación en la base de datos y utilizar el lenguaje de consulta estructurada (SQL) en el código de la aplicación para acceder a estos datos y actualizarlos. Elastic Beanstalk administra la instancia de base de datos y ayuda a mantener la conectividad entre las instancias EC2 y la base de datos.

Elastic Beanstalk crea y administra entornos escalables. Puede configurar el grupo de Auto Scaling para que escale automáticamente la aplicación para gestionar cargas masivas de tráfico. También le proporciona un nombre de dominio único para su entorno de aplicación. La sintaxis URL es <your-application>.elasticbeanstalk.com. También puede elegir su propio nombre de dominio como el nombre de dominio proporcionado utilizando Amazon Route 53.

Elegir la solución de automatización adecuada



En esta unidad, presentamos al menos cuatro servicios de AWS. Una de las preguntas más frecuente está relacionada con los diferentes servicios que ofrecen funciones de administración de aplicaciones: ¿cuál es la diferencia entre ellos o qué servicio debe utilizarse bajo qué circunstancias? Su decisión debe depender del nivel relativo de comodidad y control que necesite.

Elastic Beanstalk es un servicio de aplicaciones fácil de utilizar para crear aplicaciones web que se ejecutan en Java, PHP, Node.js, Python, Ruby o Docker. Si desea subir su propio código sin necesidad de personalizar el entorno, Elastic Beanstalk puede ser una buena elección para usted.

OpsWorks le permite lanzar una aplicación y definir su infraestructura y la especificación de cada componente, incluida la instalación de paquetes, la configuración de software y recursos (como el almacenamiento). Puede utilizar plantillas para las tecnologías más habituales (servidores de aplicaciones, bases de datos, entre otras) o puede crear su propia plantilla.

Tanto Elastic Beanstalk como OpsWorks ofrecen un nivel de servicio superior al de crear y mantener plantillas de AWS CloudFormation para crear pilas o administrar instancias EC2 directamente. Sin embargo, la elección del servicio correcto (o combinaciones de servicios) para utilizar depende de sus necesidades. Estas herramientas están disponibles para usted. Como arquitecto, debe decidir qué servicios serán los más adecuados para su caso de uso.

Estos son algunos de los aprendizajes clave de esta lección de la unidad:

- AWS Elastic Beanstalk crea y administra un entorno de aplicaciones web escalable y de alta disponibilidad que le permite centrarse en el código de la aplicación
- Puede crear su código de aplicación de Elastic Beanstalk en Java, .NET, PHP, Node.js, Python, Ruby, Go o Docker
- Los recursos de AWS creados por Elastic Beanstalk son totalmente transparentes: es decir que son visibles en las vistas de página del servicio de la consola de administración de AWS
- Sin cargo adicional para Elastic Beanstalk: solo paga por los recursos subyacentes que se utilizan



Ha llegado el momento de hacer un repaso de la unidad y concluir con una evaluación de conocimientos.

A modo de resumen, en esta unidad aprendió a hacer lo siguiente:

- Reconocer cuándo se debe implementar automatización y por qué
- Identificar cómo modelar, crear y administrar una recopilación de recursos de AWS mediante AWS CloudFormation
- Utilizar las plantillas de Quick Starts de AWS CloudFormation para configurar una arquitectura
- Indicar cómo utilizar AWS System Manager y AWS OpsWorks para la automatización de la infraestructura y la implementación