

Diseño de aplicaciones modernas con los contenedores de Linux

Contenedores: la estrategia liviana y nativa de la nube que redefine el desarrollo de aplicaciones

Conozca el contenido

03 INTRODUCCIÓN

04 UN POCO DE HISTORIA

05 CAPÍTULO 1: Nociones básicas

- 05 ¿Qué es un contenedor de Linux?
- 05 Más calidad, más velocidad y menos gastos, todo en el mismo lugar
- 06 La virtualización frente a los contenedores
- 07 Contenedores + microservicios = la combinación ideal más novedosa
- 07 Reducción, reutilización y reciclado con imágenes de contenedores

08 CAPÍTULO 2: Mejore la productividad

- 08 Los contenedores y usted
 - 08 Sea flexible
 - 08 Piense en grande
 - 08 Trabaje con más inteligencia
- 09 Logre la estandarización (real)
- 09 Escriba las aplicaciones una vez y ejecútelas donde quiera
- 09 Distribuya aplicaciones con una calidad inigualable
- 10 Utilice las herramientas y los lenguajes de su preferencia
- 10 Aumente su valor personal

11 CAPÍTULO 3: Los contenedores en el mundo real

- 11 Realojamiento "lift and shift"
- 11 Rediseño de aplicaciones
- 12 Desarrollo de aplicaciones nuevas
 - 12 Microservicios
 - 12 Aplicaciones híbridas
 - 12 Trabajos y tareas repetitivos
 - 12 Inteligencia artificial y aprendizaje automático

13 CAPÍTULO 4: Aspectos que se deben considerar y desafíos

- 13 Qué se debe tener en cuenta antes de comenzar
 - 13 Determine su estrategia de datos
 - 13 Facilite la comunicación entre los contenedores
 - 13 Lleve a cabo la sincronización y la estandarización
 - 14 Recopile todos los registros
 - 14 Mejore la seguridad
- 14 Desafíos
 - 14 Manténgase a la vanguardia de la tecnología en evolución
 - 14 Adopte la cultura DevOps
 - 14 Mantenga actualizados sus sistemas de seguridad
 - 14 Gestione y supervise los contenedores

15 CONCLUSIÓN

- 15 Información adicional
- 15 Recursos adicionales

Introducción

¿Qué contiene?

El desarrollo de aplicaciones monolíticas ya es cosa del pasado. Y aunque el término "transformación digital" parezca estar muy de moda (como sucede con los términos blockchain, ágil y nube), no es solo publicidad. La transformación ofrece niveles de velocidad, uniformidad y eficiencia sin precedentes, que cambian las bases de la forma de trabajar de los desarrolladores.

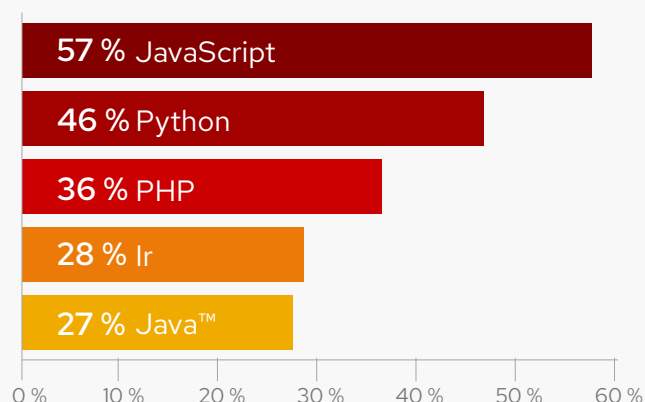
Sí, eso lo incluye a usted.

La mayor parte de la responsabilidad y la presión que conlleva cumplir con lo que prometen las empresas recae en los desarrolladores de TI. En un mundo en que los usuarios exigen aplicaciones, características y actualizaciones nuevas en todos sus dispositivos y en tiempo real, su arma secreta es la tecnología de contenedores. Los contenedores le permiten trabajar de manera más inteligente, ya que crean entornos uniformes para que en poco tiempo logre desarrollar e implementar aplicaciones nativas de la nube que puedan ejecutarse en cualquier lugar. Gracias a ellos, también puede ofrecer microservicios que eliminan los interminables ciclos de pruebas de regresión, realizar implementaciones sin interrupciones y proporcionar un mecanismo para ejecutar parches o restaurar los códigos con todas sus características.

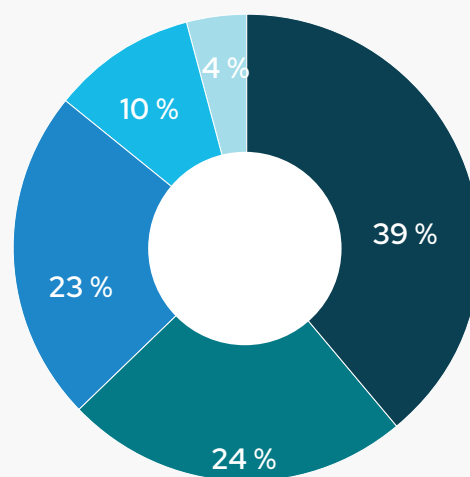
Descubra cómo implementar aplicaciones en contenedores. Regístrese en el curso de capacitación ►

Si bien los contenedores son una tecnología relativamente nueva, los desarrolladores de todo el mundo han comenzado a adoptarlos como herramientas valiosas y fundamentales para el desarrollo moderno. La rápida evolución de esta tecnología y las mejoras de las herramientas de gestión y organización de los contenedores han permitido que su adopción alcance un nuevo punto de inflexión, ya que el 49 % de los desarrolladores los utilizan en la actualidad, según un informe de Digital Ocean de 2018¹.

LENGUAJES QUE SE UTILIZAN PARA EL DESARROLLO EN CONTENEDORES¹



VENTAJAS DEL USO DE CONTENEDORES¹



- Escalabilidad rápida
- Pruebas más rápidas
- Pruebas simplificadas
- Posibilidad de evitar la dependencia de un proveedor
- Otros

¹ Digital Ocean. [Currents](#). Junio de 2018.

Un poco de historia

El proceso es el objetivo

"Con la imprenta de Johannes Gutenberg, aumentó la demanda de anteojos, ya que la práctica novedosa de la lectura permitió a los europeos darse cuenta de que sufrían de hipermetropía. A su vez, la demanda de anteojos incentivó a una cantidad cada vez mayor de personas a fabricar lentes y experimentar con ellos, lo que derivó en la invención del microscopio, que poco después nos permitió descubrir que nuestros cuerpos estaban formados por células microscópicas".

Steven Johnson

How We Got to Now: Six Innovations That Made the Modern World

Al igual que sucedió con muchos otros grandes avances tecnológicos, los contenedores son el resultado de muchos conceptos y tecnologías que evolucionaron con el tiempo. Durante las décadas de los 70 y los 80, se comenzaron a desglosar los códigos en objetos, jugando con las ideas de abstracción y aislamiento. No tardamos en darnos cuenta de que si protegíamos algunas partes del código y exponíamos otras, no solo podríamos controlar mejor el procesamiento y el manejo de datos, sino que también obtendríamos la flexibilidad necesaria para incorporar los sistemas adyacentes. Estos avances dieron lugar a nuevos elementos y procesos de abstracción e implementación de capas, y posibilitaron la evolución hacia entornos de múltiples niveles y arquitecturas orientadas al servicio (SOA), los cuales aislaron aún más las capas de datos de los códigos empresariales y las interfaces de usuario. Con el paso del tiempo, también evolucionaron nuestras metodologías, del desarrollo monolítico en cascada, al ciclo de vida de desarrollo de software (SDLC), pasando por una era de desarrollo ágil y scrum, hasta llegar finalmente a DevOps y la distribución continua.

Desde una perspectiva empresarial, todos estos avances contribuyeron a agilizar el proceso de desarrollo para generar mejores códigos, en menos tiempo y de manera más rentable. Desde el punto de vista de los desarrolladores, cada nueva iteración de innovación acorta el ciclo de desarrollo y exige una adhesión más estricta a los patrones y las metodologías. Si bien cada avance significó un paso más en la dirección correcta, la llegada de los contenedores permitió unificar todos los elementos y nos proporcionó la solución que aportaría verdadera flexibilidad, interoperabilidad y portabilidad a nuestro código.

Si observamos específicamente la evolución de los contenedores Linux®, veremos cuán significativo ha sido su impacto durante los últimos 15 o 20 años.



2000

La tecnología de contenedores apareció por primera vez como jaulas de FreeBSD, que permitía la partición de un servidor en varios subsistemas, para que los desarrolladores pudieran trabajar sin comprometer el resto del sistema.



2001

El concepto de los contenedores llegó a Linux mediante el proyecto Linux-VServer, cuya meta era ejecutar varios servidores de Linux de uso general en un único módulo.



2007

Se combinaron varias tecnologías para hacer realidad la implementación en contenedores. Específicamente, fueron los grupos de control (cgroups), systemd y los espacios de nombres del kernel o de los usuarios los que aportaron las funciones generales de control y virtualización que sirvieron de base para separar los entornos.



2008

Docker apareció en escena con su tecnología de contenedores, que incorporó aún más conceptos y herramientas para que los usuarios pudieran diseñar rápidamente nuevos contenedores en capas y compartirlos con los demás sin ningún problema.



2012

La arquitectura de microservicios evolucionó gracias a que se especializó y perfeccionó la arquitectura orientada al servicio, que se usaba para diseñar sistemas de software más flexibles, independientes y fáciles de implementar⁴.



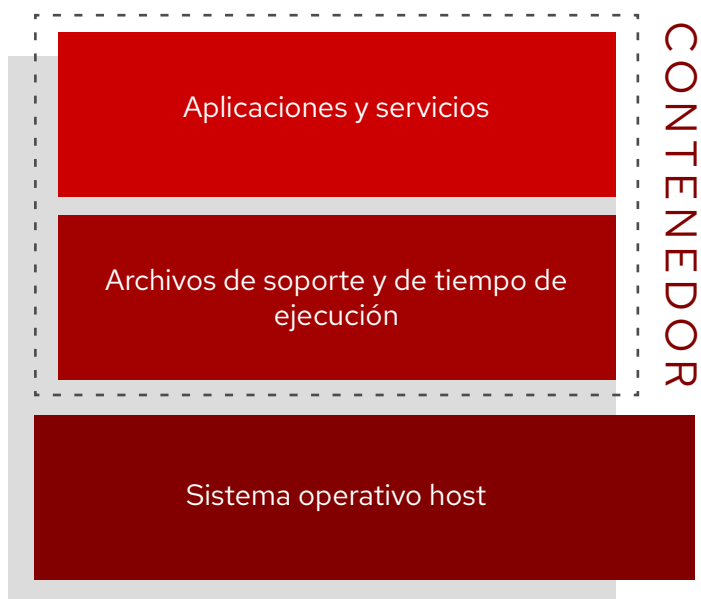
2015

Kubernetes se lanzó como un sistema de organización en contenedores de open source para automatizar la implementación, la escalabilidad y la gestión de las aplicaciones.

Capítulo 1: Nociones básicas

¿Qué es un contenedor de Linux?

Un contenedor es básicamente una nueva forma de extraer uno o más procesos del resto del sistema. Los contenedores alivianan la carga, por así decirlo, y le permiten trabajar en pequeños subconjuntos de códigos sin que esto afecte al entorno general de tiempo de ejecución. También ofrecen una forma estándar de empaquetar y aislar el código, las configuraciones y las dependencias de las aplicaciones en un solo objeto.



Más calidad, más velocidad y menos gastos, todo en el mismo lugar

La importancia de los contenedores radica en la portabilidad. Los contenedores de Linux ofrecen uniformidad y predictibilidad durante todas las etapas del proceso (del desarrollo a las pruebas, y finalmente a la producción), ya que ofrecen todos los archivos necesarios para ejecutar un elemento, una característica o una aplicación en contenedores en una imagen distinta. Gracias a ello, la implementación de contenedores es mucho más rápida, confiable y rentable que con los procesos de desarrollo monolítico, los cuales se basan en entornos de desarrollo, prueba y producción replicados. Además, puede escribir y desarrollar códigos en contenedores una sola vez, e implementarlos en diversos entornos operativos, sin que esto requiera más tiempo de desarrollo, ciclos de prueba interminables o procesos de implementación específicos.

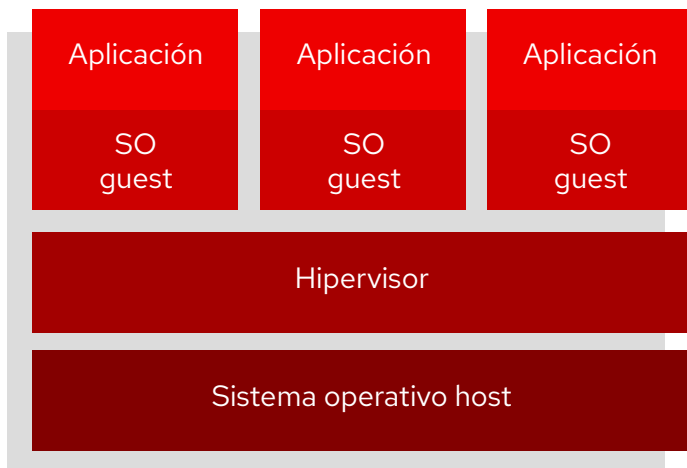
Los contenedores comparten un sistema operativo (SO) que está instalado en el servidor y se ejecutan como procesos aislados de los recursos, para garantizar una implementación rápida, confiable y uniforme, independientemente del entorno.

La importancia de los contenedores radica en la **portabilidad**.

La virtualización frente a los contenedores

Si bien los contenedores y la virtualización parecen ser conceptos similares, la realidad es que son muy diferentes.

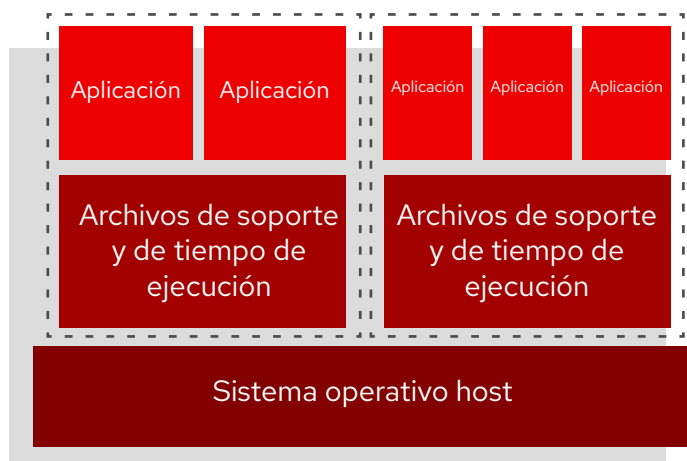
LA VIRTUALIZACIÓN



La virtualización permite que varias computadoras distintas funcionen en un mismo sistema de hardware. Los sistemas operativos y sus aplicaciones comparten los recursos de hardware desde un único servidor host. Cada máquina virtual (VM) necesita su propio sistema operativo subyacente. El encargado de crear y ejecutar dichas máquinas virtuales es el hipervisor.

EN
COMPARACIÓN
CON

LOS CONTENEDORES



El contenedor aísla los procesos de la aplicación del resto del sistema y solo incluye lo necesario para ejecutar cierta aplicación en particular, como algunos archivos del sistema operativo, bibliotecas y programas de soporte y recursos del sistema. Los contenedores son livianos, tardan mucho menos en iniciar que las máquinas virtuales y utilizan solo una parte de la memoria de ellas.

Contenedores + microservicios = la combinación ideal más novedosa

Ahora que sabe lo que es un contenedor en términos generales, analicemos la función más importante de ellos para los desarrolladores: el desarrollo de microservicios.

Los microservicios consisten en pequeñas aplicaciones autónomas con una única función, que se comunican a través de interfaces de programación de aplicaciones (API). Uno de los principios fundamentales de la arquitectura de microservicios es que cada uno de ellos ejecuta solo una función y ofrece una API bien definida que permite la comunicación dentro y fuera del código. Los microservicios son el mecanismo de encapsulación más novedoso. Dado que son completamente autónomos, realizar un cambio en uno de ellos supone menos riesgo para la aplicación en general, que cambiar el código en una estructura de código monolítica. Además, esta autonomía y su uso independiente de los recursos del sistema los vuelve más veloces y ágiles que las aplicaciones tradicionales.

Los contenedores y los microservicios pueden existir de manera independiente; de hecho, sucede a menudo. De manera individual, cumplen propósitos diferentes. Sin embargo, cuando se implementan en conjunto, se convierten en una herramienta eficaz para crear aplicaciones portátiles nativas de la nube.

Pensemos en los contenedores como una tecnología de habilitación de microservicios. Los contenedores se aíslan del sistema operativo host y contienen todos los archivos de soporte y de tiempo de ejecución necesarios para ejecutar el código que portan. Al implementar un contenedor, se ejecutará independientemente del sistema operativo subyacente. Los contenedores son portátiles, es decir, pueden implementarse en múltiples nubes y dispositivos sin tener que volver a diseñarlos ni someterlos a pruebas adicionales.

El desarrollo de microservicios con la implementación en contenedores se está generalizando cada vez más en el ámbito del desarrollo empresarial. La arquitectura ofrece niveles sin precedentes de agilidad, velocidad y manejo eficiente de los recursos para muchas de las tareas que realiza a diario en su función de desarrollador. El uso de contenedores y microservicios en un entorno de DevOps permite que los desarrolladores implementen cada servicio por separado. Esta práctica elimina la necesidad de fusionar los cambios realizados en el código, mejora notablemente el proceso de prueba y colabora con el aislamiento de fallas durante las etapas de prueba y producción. Asimismo, los equipos paralelos de desarrolladores pueden trabajar en las aplicaciones de bajo acoplamiento y elegir la pila tecnológica que mejor se adapte a sus requisitos, sin que los demás equipos tengan que cumplir con ellos.

Reducción, reutilización y reciclado con imágenes de contenedores

Si algo les fascina a los desarrolladores, es poder reutilizar el código. Con los contenedores, puede crear imágenes de contenedores de base, incorporarlas a un repositorio y descargarlas cada vez que vaya a comenzar un proyecto nuevo. Una imagen de contenedor de base es un archivo estático inmutable que no incluye los ejecutables; por lo tanto, no solo es uniforme y portátil, sino que también puede ejecutar un proceso aislado en cualquier infraestructura. La imagen consta de bibliotecas del sistema, herramientas del sistema y otras configuraciones de la plataforma que las aplicaciones necesitan para ejecutarse.

Puede crear sus propias imágenes de contenedores o buscar una en los repositorios públicos disponibles. Muchos de los proveedores de software, como Red Hat y Microsoft, crean imágenes de sus productos y las ponen a disposición de todo el mundo.

Los microservicios
y los contenedores
son una
combinación
eficaz, sobre todo
al integrarse en un
entorno de DevOps.

LOS CINCO BENEFICIOS PRINCIPALES DE LAS IMÁGENES DE CONTENEDORES

1

Automatización de los
procesos de diseño e
implementación

2

Colocación de etiquetas
para facilitar la ubicación y
la descarga

3

Simplificación del análisis
que permite detectar
puntos vulnerables

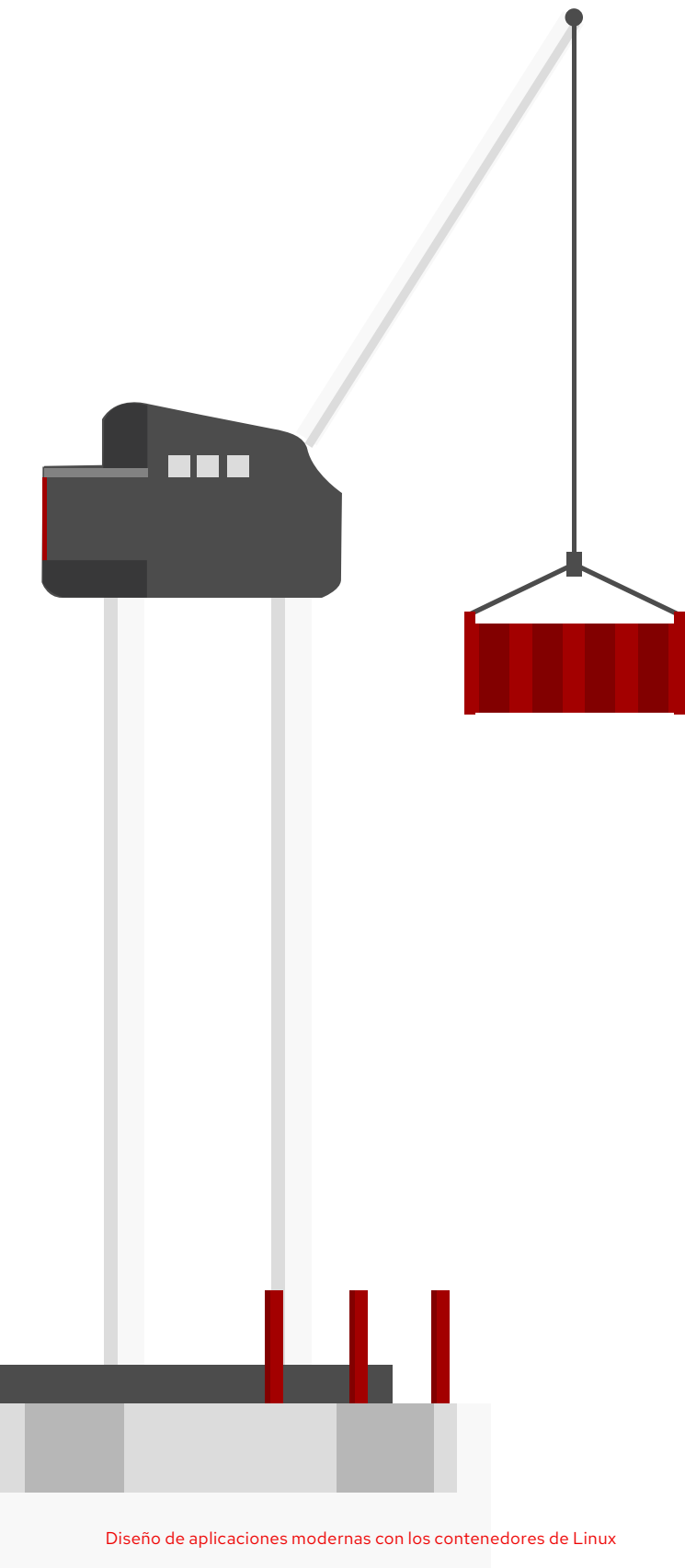
4

Cumplimiento de las
normas y las políticas de
codificación

5

Ahorro de tiempo y fomento
de la reutilización

Capítulo 2: Mejore la productividad



Los contenedores y usted

Los contenedores ofrecen un enfoque simplificado para diseñar, probar, implementar y redistribuir las aplicaciones, lo cual los convierte en una excelente opción tanto para los proyectos sencillos como para las aplicaciones más importantes. Los beneficios del uso de contenedores para toda la TI y la empresa son muchos, pero concentrémonos por un momento en cómo impulsar su éxito diario como desarrollador.

Sea flexible

Nunca hay suficiente tiempo para los ciclos de desarrollo. La empresa tiene un plan, el equipo de pruebas tiene una lista con defectos persistentes, los equipos de operaciones se esfuerzan por proteger los sistemas y ejecutar parches en ellos, y los clientes tienen sus propias ideas sobre lo que realmente importa. Desgraciadamente, el día tiene solo 24 horas, y usted tiene cierta cantidad de desarrolladores que trabajan en determinados proyectos. Cuando las aplicaciones se ejecutan en contenedores, la empresa, los equipos de operaciones y usted pueden responder de inmediato a los registros de cambios, los defectos, los problemas de seguridad, los niveles de parches y las solicitudes de funciones nuevas para los clientes. Y, además, como se puede implementar un solo contenedor sin afectar el resto de la arquitectura de aplicaciones, puede realizar cambios sobre la marcha sin necesidad de esperar a que los demás equipos terminen sus ciclos de desarrollo.

Piense en grande

Los contenedores son livianos y suelen iniciarse en cuestión de milisegundos. No requieren el arranque del sistema operativo, y solo cargan las dependencias que necesitan. También es posible crear, replicar o destruir contenedores en cuestión de segundos. Si los clientes crean cargas temporales o si la empresa decide incorporar nuevos equipos de usuarios, los equipos de operaciones podrán responder agregando los recursos que vayan a utilizar los contenedores, a medida que sea necesario. Asimismo, podrá expandir horizontalmente las aplicaciones en contenedores a los nuevos usuarios simplemente implementando nubes nuevas, incluso a nivel mundial.

Trabaje con más inteligencia

Los contenedores le permiten centrarse en la lógica de las aplicaciones sin tener que preocuparse por sus configuraciones ni por las versiones específicas del sistema operativo. Después de todo, para eso están los equipos de operaciones de TI, ¿verdad? Otra característica de los contenedores es que empaquetan el código, las dependencias y las configuraciones en una sola unidad encapsulada que se puede implementar, probar y controlar por versión con facilidad. La combinación de los contenedores con una arquitectura basada en servicios permite probar y mejorar las aplicaciones sin mayor complejidad, así como darles soporte.

Logre la estandarización (real)

Todos sabemos que la estandarización de un conjunto de entornos, desde el desarrollo hasta la producción, parece ser casi imposible. Pues bien, prepárese para ver un poco de magia. Uno de los beneficios más importantes de los contenedores es que estandarizan los entornos locales, de desarrollo, pruebas, control de calidad y producción. Gracias a este nivel de previsibilidad, podrá poner en marcha entornos aislados y pasar menos tiempo depurando y diagnosticando los problemas que se originan a causa de las diferencias en los niveles de parches, los sistemas operativos y las aplicaciones. Ahora, podrá invertir su tiempo en desarrollar y distribuir funciones nuevas. Los nuevos desarrolladores del equipo también podrán comenzar a trabajar mucho más rápido sin perder tiempo en la instalación y la configuración de sus entornos de desarrollo locales, ya que solo tendrán que descargar una imagen de contenedor de un repositorio y comenzar con el proceso de codificación.

Escriba las aplicaciones una vez y ejecútelas donde quiera

La frase "escriba las aplicaciones una vez y ejecútelas donde quiera" había tenido sus limitaciones; la más importante involucraba al sistema operativo donde se había desarrollado la aplicación. Antes podía ejecutar las aplicaciones donde quisiera, siempre y cuando el sistema de destino funcionara con el mismo sistema operativo. Gracias a los contenedores, su código puede ejecutarse prácticamente en cualquier entorno, sin importar dónde se desarrolló. Los sistemas que se desarrollan dentro de un contenedor pueden implementarse en los sistemas operativos Linux, Windows o Mac, en equipos sin sistema operativo, máquinas virtuales, nubes públicas, nubes privadas y entornos híbridos. Además, la adopción generalizada del proyecto open source de Docker ofrece una forma estable de automatizar la implementación de aplicaciones dentro de contenedores, lo cual le deja tiempo libre para que pueda pasar al siguiente ciclo de desarrollo.

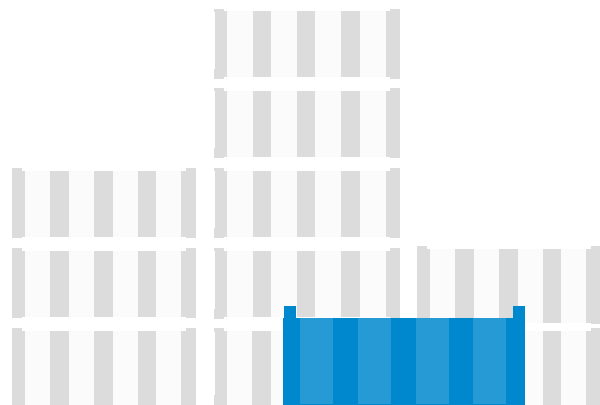
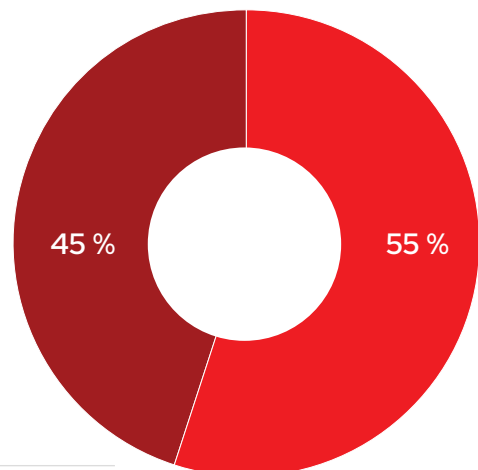
Distribuya aplicaciones con una calidad inigualable

Los contenedores agilizan y facilitan las pruebas y la resolución de problemas para todas las partes involucradas. Para usted, la capacidad de limitar el desarrollo a una sola función de la aplicación significa menos posibilidades de introducir errores involuntarios en el código adyacente. Una vez que el contenedor se somete al control de versiones, los equipos de control de calidad pueden realizar las pruebas directamente en la imagen del contenedor, en lugar de tener que extraer y diseñar toda la aplicación a partir de un servidor de integración continua (CI), lo cual simplifica las pruebas y permite ahorrar tiempo.

Los contenedores también ayudan a que los equipos de soporte y control de calidad identifiquen en menos tiempo la causa principal de algún problema en la aplicación. Debido al bajo acoplamiento del código impulsado por la API, es menor la cantidad de interdependencias que se deben revisar cuando ocurre algo inesperado. Los ingenieros pueden llegar a la raíz del problema rápidamente e identificar qué contenedor fue el responsable del error. Una vez identificado el problema, los contenedores permiten revertir los cambios con mayor facilidad. Los contenedores individuales pueden restaurarse sin perjudicar el resto de la aplicación. Cuando el contenedor vuelva a su escritorio, ya no necesitará tanto tiempo para resolver el problema, y podrá invertirlo en desarrollar las características que los líderes empresariales y los clientes desean. Con los contenedores, es posible reparar los errores el mismo día, lo cual mantiene conformes tanto a los gerentes como a los clientes.

Según un estudio reciente de IDC,

el 55 % de los líderes de TI implementó contenedores in situ, mientras que el 45 % lo hizo en la nube pública⁵.



Utilice las herramientas y los lenguajes de su preferencia

Como desarrollador, sabe lo liberador que es poder elegir las herramientas o los lenguajes que desea utilizar en un proyecto. Los contenedores brindan la flexibilidad para hacerlo. Algunos tiempos de ejecución de las aplicaciones son más adecuados para ciertos tipos de cargas de trabajo o arquitecturas. Por ejemplo, vert.x fomenta el uso de arquitecturas reactivas distribuidas, que funcionan a la perfección con aplicaciones receptivas e inmediatas, como las que se necesitan para los dispositivos de Internet de las cosas (IoT). Si bien podría diseñar este tipo de aplicaciones con otro lenguaje, tendría que reinventar lo que vert.x ofrece de forma nativa. Dado que los contenedores son realmente independientes, lo cual le brinda flexibilidad y control, puede elegir la herramienta que mejor funcione para cierta aplicación, sin que todos los equipos de aplicaciones tengan que adoptar las mismas herramientas o lenguajes para sus proyectos.

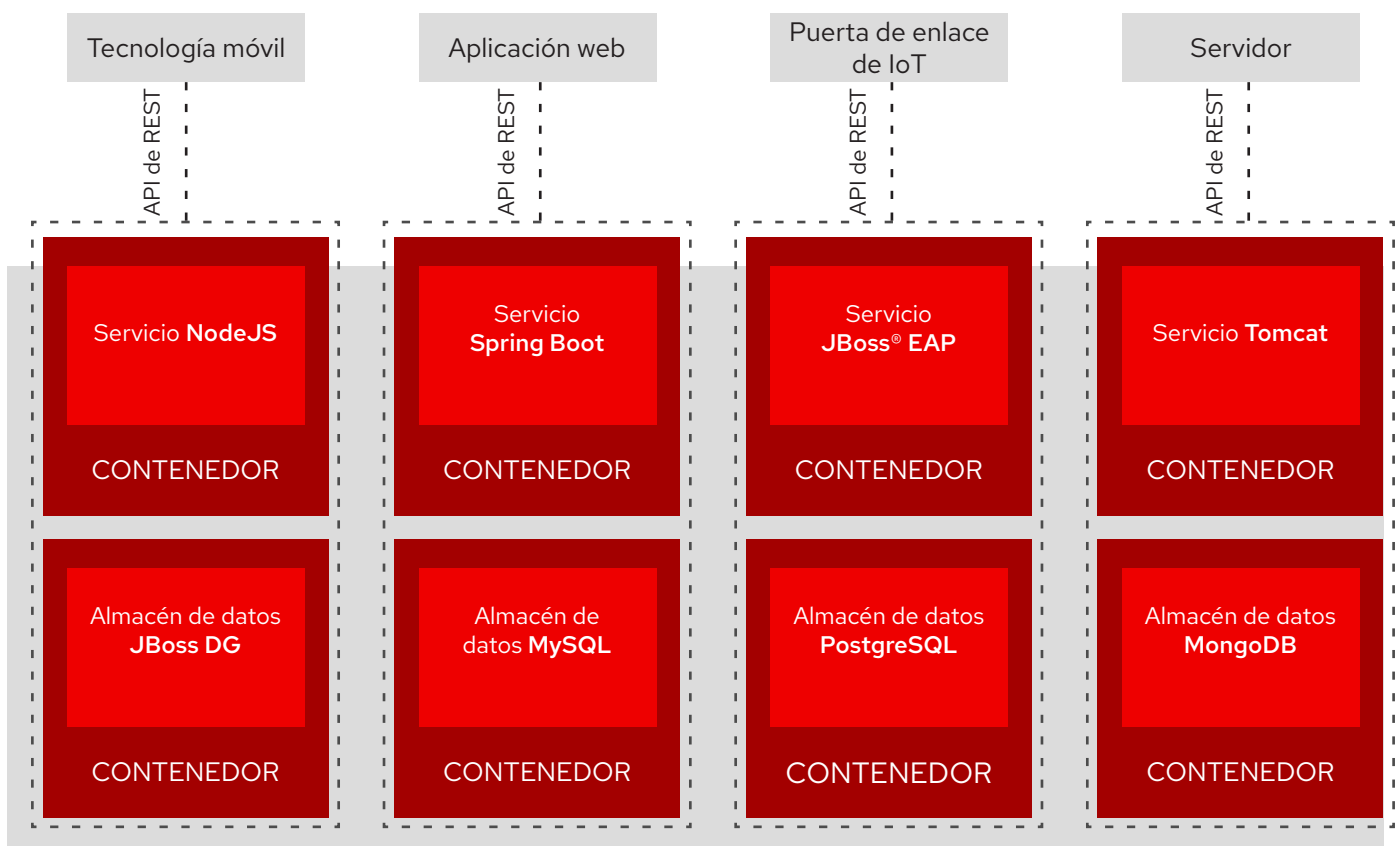


Figura 1: Aumente la flexibilidad del desarrollo y simplifique la implementación con los contenedores

Aumente su valor personal

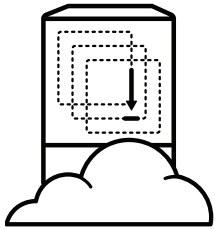
Seamos realistas con este último beneficio. Cuando se trata de su carrera, si no desarrolla sus habilidades con los avances tecnológicos, se quedará atrás o, peor aún, terminará atascado trabajando con aplicaciones heredadas hasta que se jubile. Las oportunidades de crecimiento y desarrollo son clave para el éxito. Además, los desarrolladores generalmente disfrutan aprender sobre nuevas tecnologías. Es una de las ventajas de este trabajo.

Los microservicios, los contenedores y las tecnologías de organización de contenedores le permiten ser exigente desde un punto de vista técnico y también tener una gran demanda, lo cual es una excelente combinación. Según una encuesta de Red Hat de 2018, se espera que el uso de los contenedores aumente en un 89 % durante los próximos dos años, y que se mantenga la tendencia de adopción masiva acelerada en el futuro⁶. Una vez que domine el desarrollo de microservicios basados en contenedores, surgirán nuevas oportunidades que le permitirán avanzar en su carrera y ofrecer valor a su empresa.

⁶ Dawson, Margaret. "Red Hat Global Customer Tech Outlook 2019: La automatización, la nube y la seguridad son las principales prioridades en materia de financiación". Blog de Red Hat. 18 de diciembre de 2018.

Capítulo 3: Los contenedores en el mundo real

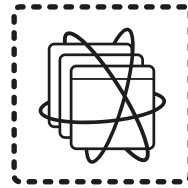
Si aún no ha recibido la solicitud de que distribuya el código en contenedores, es muy probable que la reciba muy pronto. A medida que las empresas realizan ejercicios de transformación digital, los equipos de liderazgo descubren que los contenedores permiten que las empresas de TI utilicen mejor los recursos, tanto humanos como informáticos, para reducir los costos, aumentar la eficiencia y ofrecer valor. Considere estos casos de uso mientras planifica sus proyectos, y plantee la idea de los contenedores durante su próxima reunión de equipo.



Realojamiento "lift and shift"

Si bien se habla mucho de las aplicaciones nativas de la nube, la mayoría de las aplicaciones empresariales aún son monolíticas. El término "lift and shift" se refiere al proceso de migración de las aplicaciones actuales a una arquitectura nativa de la nube más moderna. Lo que se hace es simplemente trasladar las aplicaciones actuales a la nube realizando la menor cantidad posible de cambios en el código. Por lo general, se trata de un proceso breve, el cual es el preferido de los líderes empresariales porque les permite usar las aplicaciones actuales más importantes para generar más valor y mejorar el rendimiento, sin invertir demasiado en un rediseño masivo.

Si bien el realojamiento lift and shift es sencillo y rápido, normalmente se considera una solución temporal, ya que no aprovecha al máximo las herramientas específicas de la nube. Si traslada alguna de sus aplicaciones con este proceso, en algún momento deberá volver a hacer una evaluación para asegurarse de que todavía esté prestando un servicio de valor, y para hacer recomendaciones relativas al rediseño o el reemplazo de las aplicaciones.



Rediseño de aplicaciones

El rediseño de aplicaciones ha sido un elemento básico de la modernización de aplicaciones desde que la primera de ellas se volvió obsoleta. En este caso, nos referimos específicamente a volver a diseñar una aplicación para los contenedores. Si bien este proceso es mucho más complejo que el realojamiento lift and shift, permite que la aplicación aproveche al máximo los beneficios de los entornos en contenedores y nativos de la nube.

Antes se pasaban por alto los esfuerzos de rediseño de aplicaciones debido a la complejidad y a la cantidad de tiempo que implicaban. Sin embargo, con los contenedores, dicho rediseño se transformó en un proceso gradual. Para comenzar, se modifican solo las partes más importantes de una aplicación y se realojan las demás, las cuales también se pueden rediseñar con el tiempo. Esto permite que la empresa siga funcionando mientras realiza otras mejoras, de acuerdo con su agenda. En muchos libros y blogs relacionados con los microservicios, este proceso se denomina "suprimir las estructuras monolíticas".

Por supuesto, aún es posible cambiar el diseño de las aplicaciones como se hacía antes, modificando la aplicación completa antes de lanzarla para su uso. Ambas estrategias tienen sus pros y sus contras. La ventaja más importante del rediseño gradual radica en el tiempo. Uno de los aspectos negativos es la tendencia a nunca trasladar todos los elementos a una plataforma nativa de la nube, lo que supone riesgos y genera complejidad durante la gestión.

Desarrollo de aplicaciones nuevas

Como ya sabe, los contenedores son una herramienta potente y flexible para desarrollar nuevas aplicaciones nativas de la nube. Poder desarrollar aplicaciones desde cero le permite aprovechar todos los beneficios de los contenedores, los cuales proporcionan una plataforma ideal para los microservicios, las aplicaciones híbridas, la automatización de tareas y trabajos repetitivos, y las aplicaciones del futuro, como la inteligencia artificial (IA) y el aprendizaje automático (ML).

Microservicios

Los contenedores están muy relacionados con las arquitecturas de microservicios. Las aplicaciones distribuidas y los microservicios se pueden aislar, implementar y ampliar fácilmente utilizando contenedores individuales como elementos de base.

Aplicaciones híbridas

Los contenedores le permiten estandarizar la implementación del código, lo cual facilita la creación de flujos de trabajo para las aplicaciones que se ejecutan entre entornos on-premise y de nube.

Trabajos y tareas repetitivos

Los trabajos y las tareas repetitivos, como el procesamiento por lotes y los trabajos de extracción, transformación y carga (ETL), se pueden desarrollar con facilidad en los contenedores para comenzar a trabajar rápidamente. Luego se pueden automatizar para facilitar las operaciones y ajustar de forma dinámica para satisfacer la demanda.

Inteligencia artificial y aprendizaje automático

Los contenedores ofrecen una nueva forma de diseñar e implementar aplicaciones portátiles en la nube que incorporan la inteligencia artificial y el aprendizaje automático. Estos contenedores pueden expandir rápidamente los modelos de inteligencia artificial y aprendizaje automático para satisfacer las necesidades de procesamiento de algoritmos de entrenamiento avanzados. Además, los contenedores con inteligencia artificial y aprendizaje automático se pueden implementar cerca de las fuentes de datos para mejorar el rendimiento y acortar los ciclos de entrenamiento.

Capítulo 4: Consideraciones y desafíos

Haga su vida más fácil: no pase por alto este capítulo.

Hemos hablado mucho acerca del impacto y el poder de los contenedores, pero como cualquier otra tecnología, suponen ciertos desafíos que debe tener en cuenta, porque pueden volverse problemáticos si no se abordan de forma adecuada, especialmente debido al aumento de contenedores empresariales que se implementan en la producción. Deberá estar atento a la evolución permanente de la tecnología y verificar que su empresa adopte, gestione y mantenga su arquitectura basada en contenedores de manera correcta.

Aspectos que debe tener en cuenta antes de comenzar

A estas alturas, probablemente tenga algunas ideas en mente para comenzar un proyecto nuevo o rediseñar una aplicación heredada. La mejor manera de adquirir experiencia y conocimientos con los contenedores es adoptarlos y comenzar con el proceso de codificación.

Sin embargo, deberá tener en cuenta algunos aspectos antes de comenzar. La creación de aplicaciones en contenedores supone algunos desafíos que no existen en un entorno tradicional. A continuación se incluye información sobre los aspectos que pueden afectar sus iniciativas de desarrollo. [Project Atomic](#) dio lugar a muchas de las ideas que se incluyen en esta sección. Después de terminar de leer este ebook, debería echarle un vistazo.

Determine su estrategia de datos

Si bien la mayoría son aplicaciones nativas de la nube sin estado, muchas de ellas requieren el almacenamiento de datos permanente. Los contenedores tienen un almacenamiento inmutable, es decir que los datos se pierden una vez que se detiene el contenedor. Es necesario tener esto en cuenta y diseñar las aplicaciones de manera tal que se puedan conservar los datos, independientemente del estado del contenedor.

En caso de que se necesite conservar los datos de la aplicación después de que un contenedor se deje de utilizar, puede asignar un volumen de almacenamiento a dicho contenedor. Estos volúmenes asignados se conservarán sin importar cuál sea el estado del contenedor. Los desarrolladores deben asegurarse de que el diseño de las aplicaciones les permita escribir datos en un almacén compartido. En el caso de las aplicaciones empresariales, las herramientas como Red Hat OpenShift® Container Storage proporcionan un almacenamiento

definido por software específicamente diseñado para entornos de contenedores. Red Hat OpenShift Container Storage almacena datos de forma permanente, incluso cuando los contenedores se ponen en marcha y se detienen. Además, amplía su capacidad fácilmente en todo tipo de implementaciones (sin sistema operativo, virtuales, de contenedores y de nubes) para mejorar aún más la portabilidad de los contenedores, sin limitarlos a su arquitectura de almacenamiento.

Facilite la comunicación entre los contenedores

Los elementos de las aplicaciones distribuidas deben comunicarse entre sí para lograr los flujos de trabajo. Las tecnologías de contenedores impulsan a los desarrolladores a explicitar los puntos de interconexión, y proporcionan un mecanismo que facilita la comunicación entre los contenedores con una API, lo cual es ideal para este tipo de comunicación, pero ¿qué sucede con las bases de datos?

Por lo general, las bases de datos tradicionales se comunican usando un socket a través de una red. Como el espacio de nombres de un contenedor cambia junto con su estado, este tipo de mecanismo heredado de comunicación no funciona. En las aplicaciones en contenedores, se necesita una plataforma de organización de contenedores, como Kubernetes o Red Hat OpenShift Container Platform, que facilite la comunicación de red entre los contenedores, las bases de datos y otros recursos de red.

Lleve a cabo la sincronización y la estandarización

Algunas aplicaciones en contenedores requieren la sincronización de ciertos atributos del host y el contenedor para lograr un comportamiento uniforme. Por ejemplo, piense en un servidor de registro centralizado que recibe datos de múltiples contenedores implementados en distintas regiones geográficas. La información y las marcas de tiempo del registro serían prácticamente inútiles para el equipo de operaciones si cada contenedor informara una hora diferente a la del host, sin incluir además la ubicación del servidor. Al sincronizar y estandarizar un conjunto de atributos del entorno, puede asegurarse de que los datos comunicados al almacén de datos central sean precisos, relevantes y utilizables.

Recopile todos los registros

Todas las aplicaciones deben registrar información adecuada que facilite la resolución de problemas. Si la aplicación registra las acciones, los errores y las advertencias en algún tipo de mecanismo de registro, deberá pensar cómo permitir que los usuarios obtengan, revisen y posiblemente conserven esos registros. Dado que los contenedores están separados por espacios de nombres y no pueden acceder directamente a los elementos sin sistema operativo, incluidos los discos duros locales, deberá replantearse su estrategia de registro.

La mejor forma de recopilar los registros es usando una herramienta diseñada para tal propósito. Las plataformas de organización de contenedores como OpenShift Container Storage recopilan automáticamente los datos de registro de los contenedores. Luego puede guardar esos datos en un almacenamiento permanente o seguir las pautas recomendadas sobre el envío de registros a la salida estándar o el error estándar para que puedan visualizarse en la consola de gestión central.

Mejora de la seguridad

Es indispensable que las aplicaciones en contenedores almacenen datos confidenciales, incluso las credenciales de acceso, de modo que los contenedores puedan comunicarse sin interrumpir reiteradamente al usuario. Sin embargo, esto puede suponer ciertos riesgos y exponer su aplicación a posibles riesgos de seguridad. La forma más común de transferir datos confidenciales en un contenedor es a través de variables de entorno que no son públicas. Las plataformas de organización de contenedores como Kubernetes y OpenShift Container Platform ofrecen mecanismos nativos para proteger las variables de entorno y transferir datos confidenciales a través de una aplicación en contenedores.

Retos

Manténgase a la vanguardia de la tecnología en evolución

La evolución y la expansión del ecosistema de contenedores avanzan a pasos agigantados. Este ecosistema incluye herramientas para crear, implementar, configurar, automatizar y gestionar contenedores. Los proyectos open source que respaldan este ecosistema son muy activos. Si bien esta es una señal alentadora sobre el futuro de la tecnología, puede ser difícil mantener las habilidades necesarias para distribuir aplicaciones utilizando estas soluciones. Como desarrollador, deberá estar al tanto de los proyectos open source en curso, inscribirse en programas de capacitación y mejorar sus habilidades para que pueda seguir evolucionando a la par de la tecnología de contenedores.

Adopte la cultura DevOps

Tal como se indicó al comienzo de este ebook, los contenedores están cambiando fundamentalmente la forma de desarrollar aplicaciones. Para poder responder a estos cambios, los equipos de aplicación deben adaptarse modificando sus procesos y su cultura. Para poder usar los contenedores con eficacia, las empresas deben tener una cultura

de DevOps sana, sin la separación tradicional que se formó bajo el modelo de código monolítico. Además, deben seguir y automatizar los principios de DevOps, lo que les permitirá crear un flujo de trabajo rápido, confiable, uniforme y centrado en la seguridad desde el desarrollo hasta la implementación. Como desarrollador, formará parte del proceso general, que comienza con el desarrollo, sigue con la implementación y culmina con la satisfacción del cliente. Su nueva función consiste en asumir la responsabilidad de los contenedores y garantizar que cualquier lógica de aplicación que cree se ejecute en la producción. Dado que tendrá la posibilidad de lanzar correcciones en tiempo real, se espera que depure, repare e implemente rápidamente el código para mantener su parte de la aplicación funcionando a su máximo potencial.

Mantenga actualizados sus sistemas de seguridad

No importa si trabajan o no con contenedores, la seguridad siempre es un desafío para los desarrolladores de aplicaciones. Si bien los contenedores permiten aislar las aplicaciones, lo cual es un aspecto favorable, la proliferación de contenedores en una empresa supone una nueva generación de riesgos de seguridad.

Los contenedores generalmente requieren que las aplicaciones se dividan en microservicios más pequeños, lo cual da como resultado un mayor tráfico de datos y reglas de control de acceso complejas. A medida que aumenta la cantidad de contenedores, también aumenta la posibilidad de crear controles de acceso poco estrictos entre ellos. Si no se cumplen al pie de la letra los protocolos de seguridad y acceso, se pueden generar puntos vulnerables en el entorno de producción. Además, muchas empresas usan repositorios de imágenes de contenedores, pero deben corroborar que esas imágenes cumplan con los requisitos de seguridad y conformidad de su empresa. Asegúrese de comprender cuáles son los riesgos en relación con los contenedores y cumpla siempre con las políticas de control de acceso.

Gestione y supervise los contenedores

A medida que aumenta la cantidad de contenedores implementados, puede volverse cada vez más difícil supervisarlos. La complejidad de la gestión aumenta considerablemente cuando implementa soluciones de nube híbrida, que posibilitan la ejecución de contenedores tanto en nubes públicas como privadas. Implementar un almacén de datos uniforme para acumular, analizar y solucionar los eventos que se generan a partir de todos los contenedores y las aplicaciones puede ser un desafío. Además, una vez que se registra un evento, puede ser difícil identificar dónde ocurrió el error. Afortunadamente, las nuevas tecnologías como Istio, Prometheus y Jaeger pueden ayudar a sortear algunos de estos desafíos.

CONCLUSIÓN

El uso de contenedores es la metodología preferida por casi todos los equipos de desarrollo empresarial modernizados, ya que actualmente el 91 % de los desarrolladores de nube y los gerentes de desarrollo eligen implementar contenedores en alguna capacidad on-premise⁷. Los contenedores le permiten trabajar de manera más inteligente y eficiente, con entornos de desarrollo uniformes para desarrollar y distribuir en poco tiempo aplicaciones nativas de la nube que puedan ejecutarse en cualquier parte.

El propósito de este ebook es ponerlo al día sobre los conceptos básicos de los contenedores y ayudarlo a comprender todos los beneficios que tienen para su empresa y para usted. Hay muchos recursos disponibles para ayudarlo a incorporar el conocimiento que ha adquirido aquí y aplicarlo a sus proyectos actuales. Muy pronto, podrá separar sus aplicaciones de la arquitectura, llevar su empresa a lo más alto y acelerar su propia carrera con la práctica de desarrollo más moderna, ágil y con vistas al futuro: el uso de los contenedores.

El mejor momento para empezar es ahora. Buena suerte en el recorrido que emprenderá para convertirse en desarrollador de aplicaciones nativas de la nube.

Más información

¿Le gustaría empezar a aprender cómo implementar aplicaciones en contenedores? Realice un curso gratuito en línea de Red Hat para conocer los conceptos del uso de contenedores y verlos en acción. Aprenderá a organizar las aplicaciones y los servicios en contenedores, a probarlos con Docker y a implementarlos en un clúster de Kubernetes utilizando Red Hat OpenShift Container Platform. También aprenderá a diseñar e implementar una aplicación a partir del código fuente utilizando la herramienta source-to-image de OpenShift Container Platform.

Obtenga más información y regístrese ►

Recursos adicionales

[Udemy](#)
[Stack Overflow](#)
[Docker](#)
[AWS](#)
[Google](#)
[Red Hat](#)

Acerca de Red Hat

Red Hat es el proveedor líder de soluciones de software open source empresarial en todo el mundo. Ha adoptado un enfoque basado en la comunidad para proporcionar tecnologías confiables y de alto rendimiento de Linux, nube híbrida, contenedores y Kubernetes. Red Hat ayuda a los clientes a integrar aplicaciones de TI nuevas y actuales, desarrollar aplicaciones nativas de la nube, estandarizar nuestro sistema operativo líder del sector y automatizar, proteger y gestionar entornos complejos. Sus servicios galardonados de asistencia, capacitación y consultoría convierten a Red Hat en un asesor de confianza para las empresas de la lista Fortune 500. Como partner estratégico de proveedores de nube, integradores de sistemas, proveedores de aplicaciones, clientes y comunidades open source, Red Hat ayuda a las empresas a prepararse para el futuro del mundo digital.

Copyright © 2019 Red Hat, Inc. Red Hat, Red Hat Enterprise Linux, Ansible, Ceph, CloudForms, Gluster, JBoss y OpenShift son marcas comerciales de Red Hat, Inc. registradas en Estados Unidos y en otros países. Linux® es la marca comercial registrada de Linus Torvalds en Estados Unidos y en otros países.

La marca denominativa de OpenStack y el logotipo Square O Design, ya sea juntos o separados, son marcas comerciales o marcas comerciales registradas de OpenStack Foundation en Estados Unidos y en otros países, y se utilizan con permiso de OpenStack Foundation. Red Hat, Inc. no está afiliada a OpenStack Foundation ni a la comunidad de OpenStack, y tampoco goza de su respaldo ni de su patrocinio.

Java y todas las marcas comerciales y los logotipos que se basan en Java son marcas comerciales o marcas comerciales registradas de Oracle America, Inc. en Estados Unidos y en otros países.